Sociedad Española para el
Procesamiento del Lenguaje Natural

ISSN: 1135-5948

2019 CALIDAD REVISTAS CIENTÍFICAS ESPAÑOLAS
FECYT
FUNDACIÓN ESPAÑOLA PARA LA CIENCIA Y LA TECNOLOGÍA
Nº DE CERTIFICADO: FECYT-099/2019
FECHA DE CERTIFICACIÓN: 14 de mayo 2012 (3ª convocatoria)
ESTA CERTIFICACIÓN ES VÁLIDA HASTA EL: 12 de julio 2020

Artículos

Tesis

Información General

Sociedad Española para el
Procesamiento del Lenguaje Natural

ISSN: 1135-5948

2019 CALIDAD REVISTAS CIENTÍFICAS ESPAÑOLAS
FECYT
FUNDACIÓN ESPAÑOLA PARA LA CIENCIA Y LA TECNOLOGÍA
Nº DE CERTIFICADO: FECYT-099/2019
FECHA DE CERTIFICACIÓN: 14 de mayo 2012 (3ª convocatoria)
ESTA CERTIFICACIÓN ES VÁLIDA HASTA EL: 12 de julio 2020

Comité Editorial

Consejo de redacción

ISSN:	1135-5948		
ISSN electrónico:	1989-7553		
Depósito Legal:	B:3941-91		
Editado en:	Universidad de Jaén		
Año de edición:	2020		
Editores:	Eugenio Martínez Cámara	Universidad de Granada	emcamara@decsai.ugr.es
	Álvaro Rodrigo Yuste	UNED	alvarory@lsi.uned.es
	Paloma Martínez Fernández	Universidad Carlos III	pmf@inf.uc3m.es

Publicado por: Sociedad Española para el Procesamiento del Lenguaje Natural
Departamento de Informática. Universidad de Jaén
Campus Las Lagunillas, EdificioA3. Despacho 127. 23071 Jaén
secretaria.sepln@ujaen.es

Consejo asesor

Raquel Martínez Unanue	Universidad Nacional de Educación a Distancia (España)
Ruslan Mitkov	University of Wolverhampton (Reino Unido)
Manuel Montes y Gómez	Instituto Nacional de Astrofísica, Óptica y Electrónica (México)
Lluís Padró Cirera	Universidad Politécnica de Cataluña (España)
Manuel Palomar Sanz	Universidad de Alicante (España)
Ferrán Pla Santamaría	Universidad Politécnica de Valencia (España)
German Rigau Claramunt	Universidad del País Vasco (España)
Horacio Rodríguez Hontoria	Universidad Politécnica de Cataluña (España)
Paolo Rosso	Universidad Politécnica de Valencia (España)
Leonel Ruiz Miyares	Centro de Lingüística Aplicada de Santiago de Cuba (Cuba)
Emilio Sanchís Arnal	Universidad Politécnica de Valencia (España)
Kepa Sarasola Gabiola	Universidad del País Vasco (España)
Encarna Segarra Soriano	Universidad Politécnica de Valencia (España)
Thamar Solorio	University of Houston (Estados Unidos de América)
Maite Taboada	Simon Fraser University (Canadá)
Mariona Taulé Delor	Universidad de Barcelona (España)
Juan-Manuel Torres-Moreno	Laboratoire Informatique d’Avignon / Université d’Avignon (Francia)
José Antonio Troyano Jiménez	Universidad de Sevilla (España)
L. Alfonso Ureña López	Universidad de Jaén (España)
Rafael Valencia García	Universidad de Murcia (España)
René Venegas Velásquez	Pontificia Universidad Católica de Valparaíso (Chile)
Felisa Verdejo Maíllo	Universidad Nacional de Educación a Distancia (España)
Manuel Vilares Ferro	Universidad de la Coruña (España)
Luis Villaseñor-Pineda	Instituto Nacional de Astrofísica, Óptica y Electrónica (México)

Revisores adicionales

Salud María Jiménez Zafra	Universidad de Jaén (España)
Pilar López Úbeda	Universidad de Jaén (España)
Antonio Moreno Ortiz	Universidad de Málaga (España)
Xabier Sarasola	Universidad del País Vasco (España)

Sociedad Española para el
Procesamiento del Lenguaje Natural

ISSN: 1135-5948

2019 CALIDAD REVISTAS CIENTÍFICAS ESPAÑOLAS
FECYT FUNDACIÓN ESPAÑOLA PARA LA CIENCIA Y LA TECNOLOGÍA
Nº DE CERTIFICADO: FECYT-099/2019
FECHA DE CERTIFICACIÓN: 14 de mayo 2012 (3ª convocatoria)
ESTA CERTIFICACIÓN ES VÁLIDA HASTA EL: 12 de julio 2020

Preámbulo

La revista *Procesamiento del Lenguaje Natural* pretende ser un foro de publicación de artículos científico-técnicos inéditos de calidad relevante en el ámbito del Procesamiento de Lenguaje Natural (PLN) tanto para la comunidad científica nacional e internacional, como para las empresas del sector. Además, se quiere potenciar el desarrollo de las diferentes áreas relacionadas con el PLN, mejorar la divulgación de las investigaciones que se llevan a cabo, identificar las futuras directrices de la investigación básica y mostrar las posibilidades reales de aplicación en este campo. Anualmente la SEPLN (Sociedad Española para el Procesamiento del Lenguaje Natural) publica dos números de la revista, que incluyen artículos originales, presentaciones de proyectos en marcha, reseñas bibliográficas y resúmenes de tesis doctorales. Esta revista se distribuye gratuitamente a todos los socios, y con el fin de conseguir una mayor expansión y facilitar el acceso a la publicación, su contenido es libremente accesible por Internet.

Las áreas temáticas tratadas son las siguientes:

- Modelos lingüísticos, matemáticos y psicolingüísticos del lenguaje
- Lingüística de corpus
- Desarrollo de recursos y herramientas lingüísticas
- Gramáticas y formalismos para el análisis morfológico y sintáctico
- Semántica, pragmática y discurso
- Lexicografía y terminología computacional
- Resolución de la ambigüedad léxica
- Aprendizaje automático en PLN
- Generación textual monolingüe y multilingüe
- Traducción automática
- Reconocimiento y síntesis del habla
- Extracción y recuperación de información monolingüe, multilingüe y multimodal
- Sistemas de búsqueda de respuestas
- Análisis automático del contenido textual
- Resumen automático
- PLN para la generación de recursos educativos
- PLN para lenguas con recursos limitados
- Aplicaciones industriales del PLN
- Sistemas de diálogo
- Análisis de sentimientos y opiniones
- Minería de texto
- Evaluación de sistemas de PLN
- Implicación textual y paráfrasis

El ejemplar número 64 de la revista *Procesamiento del Lenguaje Natural* contiene trabajos correspondientes a dos apartados diferentes: comunicaciones científicas y resúmenes de tesis. Todos ellos han sido aceptados mediante el proceso de revisión tradicional en la revista.

Queremos agradecer a los miembros del Comité asesor y a los revisores adicionales la labor que han realizado.

Se recibieron 28 trabajos para este número, de los cuales 24 eran artículos científicos y 4 resúmenes de tesis. De entre los 24 artículos recibidos, 13 han sido finalmente seleccionados para su publicación, lo cual fija una tasa de aceptación del 54%.

El Comité asesor de la revista se ha hecho cargo de la revisión de los trabajos. Este proceso de revisión es de doble anonimato: se mantiene oculta la identidad de los autores que son evaluados y de los revisores que realizan las evaluaciones. En un primer paso, cada artículo ha sido examinado de manera ciega o anónima por tres revisores. En un segundo paso, para aquellos artículos que tenían una divergencia mínima de tres puntos (sobre siete) en sus puntuaciones, sus tres revisores han reconsiderado su evaluación en conjunto. Finalmente, la evaluación de aquellos artículos que estaban en posición muy cercana a la frontera de aceptación ha sido supervisada por más miembros del comité editorial. El criterio de corte adoptado ha sido la media de las tres calificaciones, siempre y cuando hayan sido iguales o superiores a 5 sobre 7.

Marzo de 2020
Los editores.

Sociedad Española para el Procesamiento del Lenguaje Natural

ISSN: 1135-5948

2019 CALIDAD REVISTAS CIENTÍFICAS ESPAÑOLAS
FECYT FUNDACIÓN ESPAÑOLA PARA LA CIENCIA Y LA TECNOLOGÍA
Nº DE CERTIFICADO FECYT-099/2019
FECHA DE CERTIFICACIÓN 14 de mayo 2012 (3ª convocatoria)
ESTA CERTIFICACIÓN ES VÁLIDA HASTA EL 12 de julio 2020

Preamble

The *Natural Language Processing* journal aims to be a forum for the publication of high-quality unpublished scientific and technical papers on Natural Language Processing (NLP) for both the national and international scientific community and companies. Furthermore, we want to strengthen the development of different areas related to NLP, widening the dissemination of research carried out, identifying the future directions of basic research and demonstrating the possibilities of its application in this field. Every year, the Spanish Society for Natural Language Processing (SEPLN) publishes two issues of the journal that include original articles, ongoing projects, book reviews and summaries of doctoral theses. All issues published are freely distributed to all members, and contents are freely available online.

The subject areas addressed are the following:

- Linguistic, Mathematical and Psychological models to language
- Grammars and Formalisms for Morphological and Syntactic Analysis
- Semantics, Pragmatics and Discourse
- Computational Lexicography and Terminology
- Linguistic resources and tools
- Corpus Linguistics
- Speech Recognition and Synthesis
- Dialogue Systems
- Machine Translation
- Word Sense Disambiguation
- Machine Learning in NLP
- Monolingual and multilingual Text Generation
- Information Extraction and Information Retrieval
- Question Answering
- Automatic Text Analysis
- Automatic Summarization
- NLP Resources for Learning
- NLP for languages with limited resources
- Business Applications of NLP
- Sentiment Analysis
- Opinion Mining
- Text Mining
- Evaluation of NLP systems
- Textual Entailment and Paraphrases

The 64th issue of the *Procesamiento del Lenguaje Natural* journal contains scientific papers and doctoral dissertation summaries. All of these were accepted by a peer review process. We would like to thank the Advisory Committee members and additional reviewers for their work.

Twenty-eight papers were submitted for this issue, from which twenty-four were scientific papers and four doctoral dissertation summaries. From these twenty-four papers, we selected thirteen (54%) for publication.

The Advisory Committee of the journal has reviewed the papers in a double-blind process. Under double-blind review the identity of the reviewers and the authors are hidden from each other. In the first step, each paper was reviewed blindly by three reviewers. In the second step, the three reviewers have given a second overall evaluation of those papers with a difference of three or more points out of seven in their individual reviewer scores. Finally, the evaluation of those papers that were in a position very close to the acceptance limit were supervised by the editorial board. The cut-off criterion adopted was the mean of the three scores given.

March 2020
Editorial board.

Sociedad Española para el Procesamiento del Lenguaje Natural

ISSN: 1135-5948

2019 CALIDAD REVISTAS CIENTÍFICAS ESPAÑOLAS
FECYT
FUNDACIÓN ESPAÑOLA PARA LA CIENCIA Y LA TECNOLOGÍA
Nº DE CERTIFICADO: FECYT-099/2019
FECHA DE CERTIFICACIÓN: 14 de mayo 2012 (3ª convocatoria)
ESTA CERTIFICACIÓN ES VÁLIDA HASTA EL: 12 de julio 2020

Artículos

Tesis

Información General

Artículos

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 13-19 recibido 12-11-2018 revisado 07-01-2019 aceptado 05-02-2019

The Coruña Corpus Tool: Ten Years On

El Coruña Corpus Tool: diez años después

Anabella Barsaglini-Castro, Daniel Valcarce
University of A Coruña
{anabella.barsaglini.castro,daniel.valcarce}@udc.es

Abstract: In this paper we provide a brief introduction to a new version of the Coruña Corpus Tool. Currently available for Windows, macOS and Linux, the Coruña Corpus Tool is a corpus management tool that facilitates the retrieval of information from an indexed textual repository. Although it works like most concordance programs, its distinguishing feature is that it allows users to search for old or non-standard characters and tags in texts and metadata files, as well as to extract and export specific data for the purposes of research. With a new set of advanced search features and other recent improvements, researchers now have access to functionalities that significantly enhance the previous user experience.
Keywords: corpus management, information retrieval, software tools, concordance, Coruña Corpus

Resumen: En este artículo presentamos una breve introducción a una nueva versión del Coruña Corpus Tool. Actualmente disponible para Windows, macOS y Linux, el Coruña Corpus Tool es una herramienta de gestión de corpus que facilita la recuperación de información desde un repositorio textual indexado. Aunque funciona como la mayoría de los programas de concordancia, su característica distintiva es que permite a los usuarios buscar caracteres y etiquetas antiguos o no estándar en archivos de texto y metadatos, así como extraer y exportar datos específicos con fines de investigación. Con un nuevo conjunto de funciones de búsqueda avanzada y otras mejoras recientes, los investigadores ahora tienen acceso a funcionalidades que mejoran significativamente la experiencia previa del usuario.
Palabras clave: gestión de corpus, recuperación de información, herramientas informáticas, concordancia, Coruña Corpus

1 *Introduction*

Created as a beta version in 2007 for the Research Group for Multidimensional Corpus-based Studies in English (MuStE[1]), and released in 2012 with the publication of the *Corpus of English Texts on Astronomy*, *CETA* (Moskowich et al., 2012), the Coruña Corpus Tool (CCT) continues to be an inseparable companion to the *Coruña Corpus of English Scientific Writing* (*CC*). From its inception in 2003, the *CC* has grown and evolved. The aim of this ongoing project is the compilation of samples of scientific texts from the late Modern English period into a specialised corpus with common principles (Crespo and Moskowich, 2010; Moskowich, 2016), making possible diachronic and synchronic studies at most linguistic levels.

In parallel with the compilation of the corpus itself, we have been developing the CCT, an information retrieval platform. This tool is designed to manage, gather and query the corpus using modern information retrieval techniques. In light of the publication of the *Corpus of Historical English Texts*, *CHET* (Moskowich et al., 2019), the CCT returns with a series of improvements in efficiency and effectiveness that will immediately be noted by users. This paper presents the main features of the CCT and its recent improvements by focusing on the basic operation of the software. Our aim is, therefore, to introduce the software to researchers interested in the analysis of discourse and language, in the belief that the tool will help them to extract specific data for their research. In order

[1] https://www.udc.es/grupos/muste

ISSN 1135-5948. DOI 10.26342/2020-64-1

to provide as much detail as possible, Section 2 deals with the general characteristics of the software, and Section 3 focuses on in improvements both as a manager for corpus compilation and as an information retrieval tool for end-users. Finally, Section 4 analyses the contributions of the CCT in contrast to similar tools.

2 About the Coruña Corpus Tool

Created as a corpus management tool to facilitate the gathering of data, the CCT has been developed by the IRLab[2] group in collaboration with the MuStE research group of the University of A Coruña, Spain (Parapar and Moskowich, 2007). Its main purpose is to retrieve information from a set of compiled documents that the *Coruña Corpus* (*CC*) comprises, in order to help linguists to extract specific data for their research. To this end, all the texts in the *CC* are compiled, marked-up, encoded and stored as XML files according to the Text Encoding Initiative P5 standard (see The TEI Consortium (2019) for the specification). In combination with the textual documents, MuStE also compiles metadata files that offer extra information about the authors and their texts.

The CCT is a multi-platform desktop application (it can be executed in Linux, macOS and Windows) written in Java. The tool parses XML files following the TEI standard and extracts the tagged fields that we wish to index, such as information about authors, date, scientific field (or the subcorpus to which a particular sample belongs), content, and document identifier. The XML files are validated using a Document Type Definition (DTD), a file that defines a set of syntax rules for XML documents. At this stage during the compilation, the tool also shows any errors found in the XML files to help coders to deal with any issues of syntax that might arise.

The tool builds a multi-field index structure that allows searches using different criteria, both in the samples and in the metadata accompanying them. Thus, users can execute queries on the whole set of documents (an option that is shown by default); on the individual document level, which allows the selection of a single sample from the whole corpus of corpora that have been loaded; or even on a subset of samples. Search results are displayed in a table or grid, typical of concordance programs, showing the document identifier, plus the position and surrounding context of the match. Advanced search features are also available. As the texts compiled date from the eighteenth and nineteenth centuries, users can use wildcards to specify spelling variants of the same form (*e. g.,* ⟨e⟩, ⟨æ⟩, ⟨œ⟩) and regular expressions to make complex queries that match patterns such as prefixes or suffixes. Additionally, they can use phrase queries involving combinations of words with a specified number of spaces between them as a means of finding specific expressions or verbal forms. Thus, if none (0), one (1) or two (2) spaces are selected when searching for “the answer”, results will include occurrences matching these consecutive terms, a term between them (*e. g.,* “the right answer”), or two elements in between them (*e. g.,* “the logical right answer”), respectively. These advanced search functionalities are implemented using Apache Lucene, an open-source information retrieval library that offers state-of-the-art search features.

The CCT is also able to generate frequency lists from the whole set of documents or a subset of the corpus or corpora (when more than one corpus is loaded). These alphabetically sorted lists also contain the number of occurrences or tokens of each term (type). Additionally, the user can use filters to select documents that satisfy the required criteria.

Finally, the tool also provides styled document rendering to view text samples and metadata files. The XML files are rendered using Cascading Style Sheets (CSS) and an integrated web browser offers a visually pleasant user experience.

The CCT is composed of two executables, called ‘Manager’ and ‘Client’. The Manager is used by compilers to build a corpus from a set of documents and metadata files, whereas the Client (available for users in general) is intended for searching the corpus, viewing documents and generating word lists and concordances.

3 New features and improvements

Over its ten years of existence, the CCT has evolved from being an Information Retrieval platform accompanying an indexed repository of English scientific texts, to a more ma-

[2] `https://www.irlab.org`

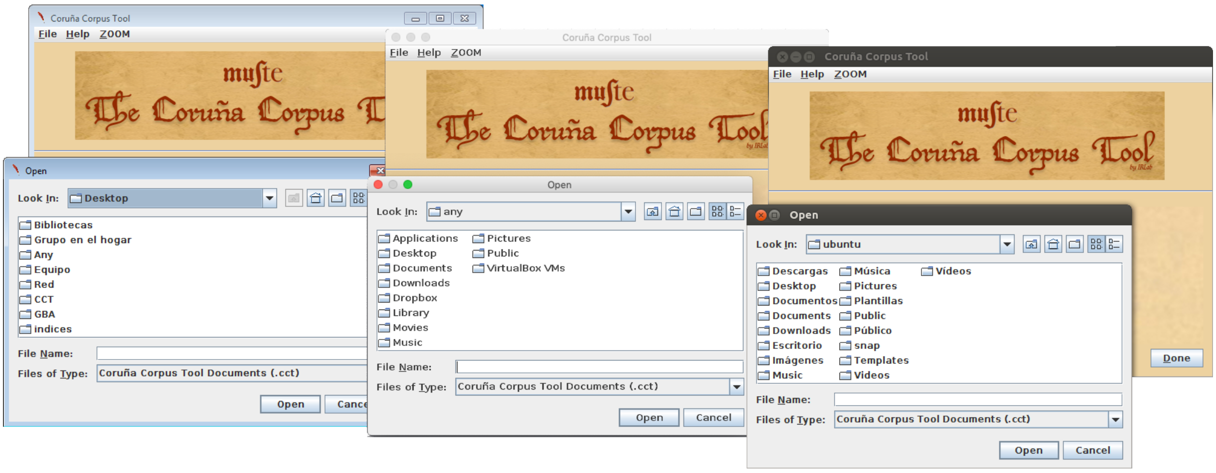


Figure 1: New "Look & Feel" on the CCT for Windows, macOS and Linux

ture kind of software that can create, manage and query the stored collections following the TEI standard (The TEI Consortium, 2019). Over the years, new functionalities have steadily been added, as well as bug fixes and improvements to usability. Nonetheless, it is important to note that compatibility is one of the fundamental principles of the development of this software. Hence, newer versions of the CCT are always backwards compatible with corpora indexed by older versions, as with the *Corpus of English Texts on Astronomy*, *CETA* (Moskowich et al., 2012) and the *Corpus of English Philosophy Texts*, *CEPhiT* (Moskowich et al., 2016). In the following subsections, we present an overview of the most notable changes.

3.1 General improvements

We have updated the CCT to work with modern versions of Java, with Java 8 being the minimum version required to run the tool. We have taken advantage of the new functionalities provided by Java 8 to improve the efficiency and security of the CCT. We also packaged all the external dependencies accompanying the older version in the same file using Maven, a Java build tool, which also offers faster compilations. In this way, users only need to interact with two applications ("manager.jar" and "client.jar" for the Manager and the Client, respectively) to access all functionalities.

We have endeavoured to solve all compatibility issues and to ensure that Windows, macOS and Linux users can all use the tool seamlessly. One of the tasks undertaken in this respect was the unification of the "Look & Feel" of the tool for the three platforms, so as to offer the same visual experience to all users (Figure 1).

3.2 User experience improvements

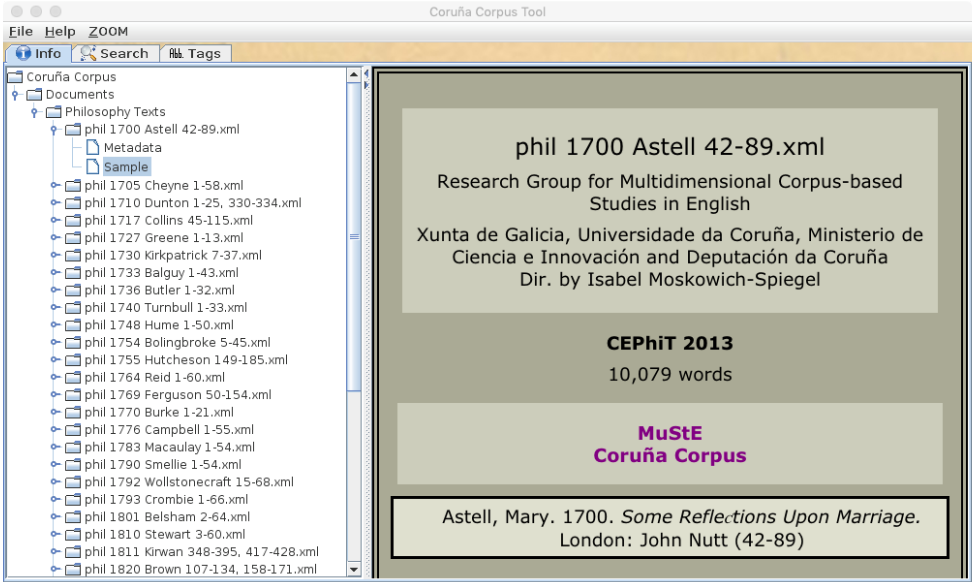
In this version of the CCT, we have improved the visualisation of documents (samples) by employing a modern web engine (Figure 2). As noted above, the XML files are styled using Cascading Style Sheets (CSS) and the new embedded web browser is capable of rendering the documents in high quality.

One feature that sets the CCT apart from other concordance programs is that from the first version it has always been designed with visual accessibility in mind. It includes a zoom feature to adapt the rendering size of the content. In the current revision of the tool, we have also added a more fine-grained control of the zoom level. Moreover, the state of the application is now saved to avoid losing the search results when the zoom level is modified.

The names of some labels have been updated, as has the size of the windows required to adapt better to modern screen resolutions. Nonetheless, users can resize the windows to make them larger if required. Finally, we have added a searching box to the digital manual to enable users to look for specific help.

3.3 Manager improvements

After several years using the Manager to index linguistic corpora, we found that the most common problems were related to file encoding. Therefore, to avoid issues of this kind entirely, the Manager encodes all XML



Coruña Corpus
Documents
Philosophy Texts
phil 1700 Astell 42-89.xml
Metadata
Sample
phil 1705 Cheyne 1-58.xml
phil 1710 Dunton 1-25, 330-334.xml
phil 1717 Collins 45-115.xml
phil 1727 Greene 1-13.xml
phil 1730 Kirkpatrick 7-37.xml
phil 1733 Balguy 1-43.xml
phil 1736 Butler 1-32.xml
phil 1740 Turnbull 1-33.xml
phil 1748 Hume 1-50.xml
phil 1754 Bolingbroke 5-45.xml
phil 1755 Hutcheson 149-185.xml
phil 1764 Reid 1-60.xml
phil 1769 Ferguson 50-154.xml
phil 1770 Burke 1-21.xml
phil 1776 Campbell 1-55.xml
phil 1783 Macaulay 1-54.xml
phil 1790 Smellie 1-54.xml
phil 1792 Wollstonecraft 15-68.xml
phil 1793 Crombie 1-66.xml
phil 1801 Belsham 2-64.xml
phil 1810 Stewart 3-60.xml
phil 1811 Kirwan 348-395, 417-428.xml
phil 1820 Brown 107-134, 158-171.xml

phil 1700 Astell 42-89.xml
Research Group for Multidimensional Corpus-based Studies in English
Xunta de Galicia, Universidade da Coruña, Ministerio de Ciencia e Innovación and Deputación da Coruña
Dir. by Isabel Moskowich-Spiegel

CEPhiT 2013
10,079 words

MuStE
Coruña Corpus

Astell, Mary. 1700. *Some Reflections Upon Marriage.*
London: John Nutt (42-89)

Figure 2: Visualization of XML samples

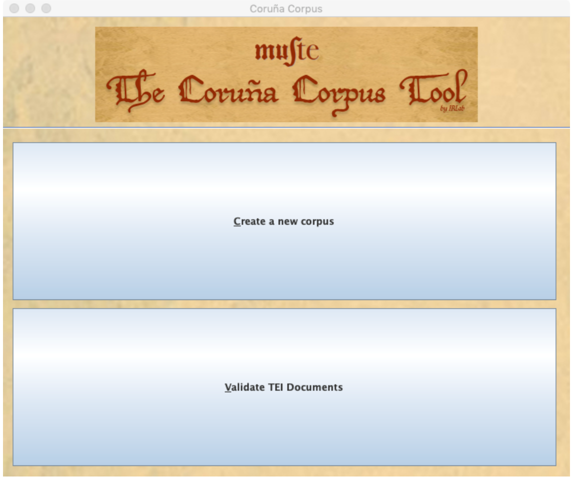


Figure 3: Manager's simplified features

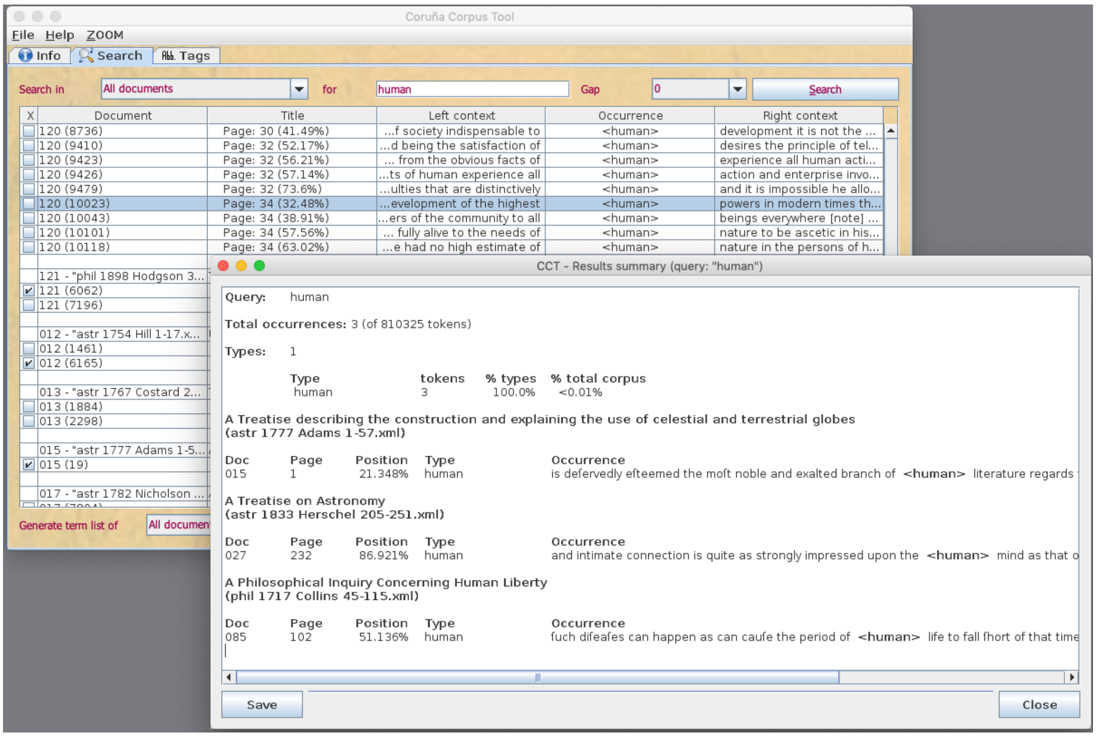
files in UTF-8 format prior to indexing; UTF-8 is a Unicode encoding standard with worldwide compatibility (The Unicode Consortium, 2019). The Manager also removes the Byte Order Mark (BOM) if it is found in a file. This mark is discouraged with UTF-8 encoding (The Unicode Consortium, 2019) and can lead to compatibility problems on some platforms. Before indexing the files, during the validation process, the Manager performs these encoding tasks automatically and the compiler is informed of the result.

Additionally, the updated Manager now uses a personalised identifier for each text. When indexing, the Manager extracts the "idno" (identification number) field from the XML files —as defined by the TEI Consortium (The TEI Consortium, 2019). These identifiers or permanent codes not only facilitate the comparison between occurrences in a search, but also allow for distinguishing between one text and another regardless of the specific corpus to which each one belongs, that is, within the *CC* as a whole, and not within each index, as was the case in previous versions. In this way, documents can be referenced using unique identifiers if compilers specify the desired identifiers in each XML file.

Since previous functionalities, such as "Manage existing corpus" and "Browse existing corpus", were no longer required for the creation of new indexes, we have also simplified the options of the Manager by keeping the two main features: the creation of a new corpus and the validation of TEI documents (Figure 3).

3.4 Client improvements

With the new version of the tool, users can search several corpora at the same time. To load a new corpus when one is already loaded, users can decide to either replace or combine the current corpus with the new one. We have added warning messages to inform the user about the status of the current loaded corpora, as well as alerts that appear when the loading process is taking place. These messages prevent unintentional actions that might provoke undesirable behaviour. This new utility of the CCT allows users to make searches across multiple cor-



X	Document	Title	Left context	Occurrence	Right context
□	120 (8736)	Page: 30 (41.49%)	...f society indispensable to	<human>	development it is not the ...
□	120 (9410)	Page: 32 (52.17%)	...d being the satisfaction of	<human>	desires the principle of tel...
□	120 (9423)	Page: 32 (56.21%)	... from the obvious facts of	<human>	experience all human acti...
□	120 (9426)	Page: 32 (57.14%)	...ts of human experience all	<human>	action and enterprise invo...
□	120 (9479)	Page: 32 (73.6%)	...ulties that are distinctively	<human>	and it is impossible he allo...
□	120 (10023)	Page: 34 (32.48%)	...evelopment of the highest	<human>	powers in modern times th...
□	120 (10043)	Page: 34 (38.91%)	...ers of the community to all	<human>	beings everywhere [note] ...
□	120 (10101)	Page: 34 (57.56%)	... fully alive to the needs of	<human>	nature to be ascetic in his...
□	120 (10118)	Page: 34 (63.02%)	...e had no high estimate of	<human>	nature in the persons of h...
	121 - "phil 1898 Hodgson 3...				
☑	121 (6062)				
□	121 (7196)				
	012 - "astr 1754 Hill 1-17.x...				
□	012 (1461)				
☑	012 (6165)				
	013 - "astr 1767 Costard 2...				
□	013 (1884)				
□	013 (2298)				
	015 - "astr 1777 Adams 1-5...				
☑	015 (19)				
	017 - "astr 1782 Nicholson ...				

```
Query:     human

Total occurrences: 3 (of 810325 tokens)

Types:     1

           Type              tokens   % types   % total corpus
            human             3         100.0%    <0.01%

A Treatise describing the construction and explaining the use of celestial and terrestrial globes
(astr 1777 Adams 1-57.xml)

Doc        Page      Position   Type                 Occurrence
015        1          21.348%   human                is deſervedly eſteemed the moſt noble and exalted branch of  <human>  literature regards

A Treatise on Astronomy
(astr 1833 Herschel 205-251.xml)

Doc        Page      Position   Type                 Occurrence
027        232        86.921%   human                and intimate connection is quite as strongly impressed upon the  <human>  mind as that o

A Philosophical Inquiry Concerning Human Liberty
(phil 1717 Collins 45-115.xml)

Doc        Page      Position   Type                 Occurrence
085        102        51.136%   human                ſuch diſeaſes can happen as can cauſe the period of  <human>  life to fall ſhort of that time
```

Figure 4: “Results Summary” window

pora files (or across any specified subset of documents) simultaneously. They can also generate frequency lists from documents from different corpora. To implement this feature, we had to change the internal architecture of the information retrieval engine, again using Lucene multi-search capabilities, to launch queries over multiple indexes. This feature had been repeatedly demanded by linguists because they wanted to conduct contrastive analyses between different subcorpora of the *CC* without having to close and reopen the application to load a different corpus whenever they needed to perform a search (or even to request personalised indexes), as was the case with the previous version of the tool. Thanks to this, researchers can now get results from the different corpora loaded (*i. e.,* more than one at the same time) and can easily visualise and compare them.

As Figure 4 illustrates, we have also added checkboxes to the search results display, to allow users to mark the occurrences on which they want to compute the summary statistics (*i. e.,* the “Results Summary” window). If no checkbox is checked, summary statistics are computed over all occurrences, as in the previous version. In addition, we have included an “unmark all” button (at the top of the frame) to uncheck all checkboxes of selected examples without having to scroll through the results window manually before saving results in the desired format.

Prior to this new version, when users clicked on a search result, the document was displayed in plain text only, in a new window. Now, users can also click on the “View” button to show the document with a more appealing rendering.

We have improved the save options of the CCT to export the desired search occurrences or summaries to an external file. All the files are encoded in UTF-8 to avoid formatting errors on any platform. These files can be opened with text editors as well as spreadsheets applications such as those provided in the LibreOffice or Microsoft Office suites.

4 Related work

To provide a brief overview of the main features of the CCT in contrast to similar software in the field, we will compare it with some well-known and web-based tools such as CQPweb (Hardie, 2012) and Wmatrix (Rayson, 2009), as well as with AntConc (Anthony, 2019).

CQPweb (Hardie, 2012) is a web-based corpus analysis system that provides a graphical user interface (GUI). Especially useful for large corpora, it is compatible with any corpora and allows word-level annotation and text-level metadata. Moreover, it is also available as open-source software. Likewise, Wmatrix (Rayson, 2009) provides a user-friendly web interface that offers the analysis of word frequency lists, keywords in context (KWIC) concordances and collocations, as well as the comparison of specific domains with larger corpora.

In contrast to these web-interface tools that allow corpora storage in a web server database, AntConc (Anthony, 2019) family tools comprise a series of downloadable and executable freeware additional software such as file converters, corpus analysis toolkits (for concordancing and text analysis for different languages) and even taggers that can be directly used without any installation or Internet connection requirements. In addition to that, AntConc provides greater flexibility in the use of corpora because it is not linked to a particular corpus. However, although all these features might represent a considerable advantage for quick searches and analysis of data, the software-corpora connection should be considered. Thus, the same way that the software designed for a specific operating system proves to be more functional and stable than third-party software, the link between the CCT and its corpora provides much more accurate and reliable search results[3].

Although the CCT works like most concordance programs by allowing basic search by single terms, concordance generation (KWIC), regular expressions search, with or without term-distance specification or wildcard searches, as well as word frequency lists and searches among several corpora or within the same corpus, it also offers some special features such as the representation of original spellings (<æ>, <œ>, <ſ>), searches discriminating spelling variants (<e>, <æ>, <œ>, or <s>, <ſ>, etc.), and the possibility to select subsets of samples by using socio-external variables such as age or sex of the author and genre of the sample, among others. In the same line to AntConc, the CCT is a free and open-source software that can be downloaded and installed on any computer (Windows, macOS and Linux), by avoiding the first-time register requirements, licenses agreements and even subscription fees that web-based services might have. Thus, despite the fact of its installation requirements or even the necessity of loading a previously indexed corpora to work with, the CCT offers a series of options that the aforementioned tools lack such as the possibility of adjusting the zoom according to the users' visual needs and a more user-friendly display of the samples and metadata. Furthermore, users can select and export their searches results into a wider variety of formats (txt, docx, and xlsx, respectively) to work with the data without needing the tool nor a network connection.

Another characteristic shared by the CQPweb (Hardie, 2012), Wmatrix (Rayson, 2009) and the CCT —and questioned by Anthony (2013, pp. 153)— is the inability users have "to observe the raw data directly with their own eyes", due to the fact of being indexed, and hence, requiring the tool to be visualized and make use of it. Although the CCT does not provide an entire solution to this respect, it does allow the visualization of the texts in two ways. Parallel to the default display that these tools tend to offer (i.e., in txt format), the CCT also provides a clear and digitalised version of the original samples, giving the user the opportunity of accessing and reading the texts without being affected by any filtering effect the tool might cause.

Overall, despite some limitations that could not been implemented yet, the CCT combines some of the best features most used in this field, with powerful and user-friendly functionalities that represent a before and after in its use.

5 *Final remarks*

In this paper, we have described the new features of the Coruña Corpus Tool. This brief overview has been intended to clearly illustrate the main characteristics and functionalities of the tool, and thus to allow users to take advantage of its current full potential. This software is extensively used by the MuStE group for the management and indexation of linguistic corpora, but also by users

[3] As such, this does not imply that the CCT is not compatible with other corpora. It is simply a factor that facilitates the analysis and accuracy of results. An example of this can be found in the frequency lists generated by those tools. Thus, the greater the software-corpus linkage is, the more accurate the results will be. Otherwise, and due to the fact that certain tools do not filter punctuation marks, the total word count is increased, forcing the researcher to perform a manual normalisation of frequencies to provide a reliable analysis.

more generally for the study of the historical development of English, for specific purposes and from different perspectives. The new features that the IRLab group has been developing in collaboration with the MuStE group at the University of A Coruña will enable linguists to easily obtain reliable data for their research and improve their user experience.

In the future, we plan to continue working on new functionalities that will improve search filters and the recovery of special characters, as well as the distinction between formulas, subscripts, and certain other elements included in scientific texts, by using specific labels. Likewise, we aim to provide a more advanced and varied display of results and to facilitate the accessibility of the tool from other devices, as well as its portability and compatibility with other corpora and/or platforms.

Acknowledgements

The research reported here has been funded by the Spanish Ministry of the Economy, Industry and Competitiveness (MINECO), grant number FFI2016-75599-P. This grant is hereby gratefully acknowledged. The second author also acknowledges the support of the Spanish Ministry of Science, Innovation and Universities, grant number FPU014/01724.

References

Anthony, L. 2013. A critical look at software tools in corpus linguistics. *Linguistic Research*, 30(2):141–161.

Anthony, L. 2019. AntConc (Version 3.5.8) [Computer Software]. Tokyo, Japan: Waseda University. Available from `https://www.laurenceanthony.net/software`.

Crespo, B. and I. Moskowich. 2010. CETA in the Context of the Coruña Corpus. *Digital Scholarship in the Humanities*, 25(2):153–164.

Hardie, A. 2012. CQPweb — combining power, flexibility and usability in a corpus analysis tool. *International Journal of Corpus Linguistics*, 17(3):380–409.

Moskowich, I. 2016. Philosophers and scientists from the Modern Age. In I. Moskowich, G. Camiña Rioboo, I. Lareo, and B. Crespo, editors, *'The Conditioned and the Unconditioned': Late Modern English texts on philosophy.* John Benjamins, Amsterdam, chapter 1, pages 1–23.

Moskowich, I., G. Camiña Rioboo, I. Lareo, and B. Crespo. 2012. *Corpus of English Texts on Astronomy.* John Benjamins, Amsterdam.

Moskowich, I., G. Camiña Rioboo, I. Lareo, and B. Crespo. 2016. *Corpus of English Philosophy Texts.* John Benjamins, Amsterdam.

Moskowich, I., B. Crespo, L. Puente-Castelo, and L. M. Monaco. 2019. *Corpus of History English Texts.* Universidade da Coruña, A Coruña.

Parapar, J. and I. Moskowich. 2007. The Coruña Corpus Tool. *Procesamiento del Lenguaje Natural*, 39:289–290.

Rayson, P. 2009. Wmatrix: a web-based corpus processing environment. *Computing Department, Lancaster University.*

The TEI Consortium. 2019. *TEI P5: Guidelines for Electronic Text Encoding and Interchange.* Text Encoding Initiative Consortium, Charlottesville, VA.

The Unicode Consortium. 2019. *The Unicode Standard, Version 12.1.0.* The Unicode Consortium, Mountain View, CA.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 21-28 recibido 29-10-2019 revisado 09-01-2020 aceptado 10-01-2020

Detecting Influencers in Social Media using information from their followers

Detectando Influencers en Medios Sociales utilizando la información de sus seguidores

Javier Rodríguez-Vidal, Julio Gonzalo and Laura Plaza
Universidad Nacional de Educación a Distancia (UNED)
{jrodriguez,julio,lplaza}@lsi.uned.es

Abstract: Given the task of finding influencers of a given domain (i.e. banking) in a social network, in this paper we investigate (i) the importance of characterizing followers for the automatic detection of influencers; (ii) the most effective way to combine signals obtained from followers and from the main profiles for the automatic detection of influencers. In this work, we have modeled the discourse used in two domains, banking and automotive, as well as the language used by the influencers in such domains and by their followers, and used these Language Models to estimate the probability of being influencer. Our most remarkable finding is that influencers not only depend on their expertise on the domain but also on that of their followers, so that the more knowledge and number of experts among their followers, the more probability of being influencer a profile has.

Keywords: Learning to Rank, Web and social media search, Information extraction, Social Network Analysis, Natural Language Processing, Social Media Influencers

Resumen: Dada la tarea de encontrar influencers en un dominio dado (i.e. banking) en una red social, en este artículo investigamos (i) la importancia de caracterizar a los seguidores para la detección automática de influencers; (ii) la manera más efectiva de combinar señales obtenidas de los seguidores y de los perfiles principales para la detección automática de influencers. En este trabajo, hemos modelado el discurso usado por los usuarios en dos dominios, automotive y banking, así como el lenguaje utilizado por los influencers en dichos dominios y por sus seguidores, y utilizamos estos Modelos de Lenguaje para estimar la probabilidad de ser un influencer. Nuestro mayor descubrimiento es que los influencers no sólo dependen de su conocimiento sobre el dominio sino del de sus seguidores; por lo tanto, cuanto mayor conocimiento y número de expertos haya entre sus seguidores, mayor será la probabilidad que el perfil sea de un influencer.

Palabras clave: Learning to Rank, Búsqueda Web y en Medios Sociales, Extracción de Información, Análisis de Redes Sociales, Procesamiento de Lenguaje Natural, Influencers en Redes Sociales

1 Introduction

In traditional marketing it is imperative to know who is talking about an entity. Opinions of anonymous people do not have the same impact as opinions of special users, well-known people within communities, and who have the power to change the opinions of other users. These kind of users are known as *influencers* or *opinion-makers*.

Before the advent of Social Media, people with the capacity of influencing the public opinion in a given domain were few and easy to identify: journalists from mass media, authorities with academic degrees and proved expertise, politicians, media owners, celebrities, etc. In practice, editorial boards and lobbies could effectively decide what information and what opinions reached the masses, and how. Public Relations (PR) for organizations and individuals were, then, a matter of addressing a few opinion makers to shape their reputation, i.e., how their image was projected to the public opinion. Social Media has significantly complicated matters for organizations from the point of view of Public Relations. Monitoring and managing social

ISSN 1135-5948. DOI 10.26342/2020-64-2

media brings unprecedented opportunities to know and interact with clients and stakeholders, but it renders previous PR methodologies obsolete. One of the key aspects of Online Media, and of Social Media in particular, is that any citizen is a candidate to become influential: it is no longer possible to narrow the filter to media owners, journalists, academic experts and other standard profiles. In this context, one of the key aspects of Online Reputation Monitoring (ORM) is to detect which social media profiles have the capacity of influencing the public opinion and, therefore, creating opinion and shaping the reputation of organizations, companies, brands and individuals (Madden y Smith, 2010).

Just as there are profiles of influencers, there are also other kind of users in Social Networks that support them and serve as a loudspeaker for the propagation of the ideas of influencers, they are called *followers*. They are in charge of spreading the ideas of an influencer, either by retweeting a post or by generating new texts from influencers' ideas.

For this reason, in this work we want to check if we can detect influencers using information from followers. In particular, we want to know if the language used by followers is an important factor for establishing whether or not a user is an *influencer*. We distinguish here two different characteristics that define influencers: (i) the possession of knowledge about the *domain* of study (i.e. brokers in economy, mechanics in automotive, etc.) or (ii) the capacity of persuade other people (*authority*) because they are well-known for the general public, for example celebrities, sportsmen, etc.

To verify our hypothesis and methods, we used data extracted from Twitter. There are many important reasons, from the point of view of ORM, to use this social network rather than others, to name a few: (i) it is *immediate*, breaking news appear and propagate first than in other social network; (ii) it is *global*, accessible in the whole world and (iii) it is *asymmetric*, there is no need to have the consent to follow an account.

2 Related Work

Influencers are a special kind of users in Social Networks. They are trustworthy to the members of their communities and their ideas are capable to change other people's mind about an entity, even jeopardizing the entity's reputation. Aral y Walker (2012) defend that, in order to predict the propagation of actions, it is important to use jointly the influence, the susceptibility and the likelihood of spontaneous adoption in the local network around individuals. But, as the authors point out, it is not clear whether influence and susceptibility are general signals or depend on the domain.

Detecting *influencers* in ORM has two distinguishing signals: first, the number of influencers is orders of magnitude lower than the number of non-influencers. Second, potential influencers are usually scanned by reputation experts, which use automatic filters as a preliminary step. Both signals are characteristic of search problems, where ranking is the most natural way of presenting results to the users (in this case, the reputation experts). This approach is followed by the RepLab campaign (CLEF, 2014), an evaluation forum for ORM that, in its 2014 edition, included the *author ranking* task. This task aimed to distinguishing the users with the most reputational influence from the less influential users. In our work, we address the author ranking task and use for our experiments an extension of the RepLab 2014 dataset, which is described in Section 3.3.1. In the following lines, we describe systems that participated in the competition.

The best system in the competition was (AleAhmad et al., 2014), which implemented the idea that people who are opinion makers will talk more about hot topics. Another approach proposed in the RepLab competition (Cossu et al., 2014) assumed that influencers tend to produce more opinionated content in tweets. (Vilares et al., 2014) used the confidence provided by the LibLinear classifier to rank the users according to their level of influence, higher confidence means higher influence. (Villatoro-Tello et al., 2014) used techniques for signal extraction and collected the most representative signals from each user's activity domain. The last participant (Lomena y Ostenero, 2014) used a small set of features based on the information that can be found in the text of tweets: POS tags, number of hashtags or number of links.

Subsequent to this competition, new studies have appeared that have worked on this dataset. Cossu, Labatut, y Dugué (2016) tested different signals and concluded that users from particular domains behave and write in their own specific way and using only text-based signals is enough to detect domain influencers. Nebot et al. (2018) used embeddings to represent each document. Rodríguez-Vidal et al. (2019) used Language Models to compute signals that model the degree of authority and domain knowledge of the profiles.

None of these previous studies have exploited the information related to the followers, more than using the number of followers signal provided by the Social Networks. In our work, we model the language that followers of a given profile use to communicate their ideas, to estimate their probability of being an authority and domain expert. Our hypothesis in this work is that the probability of being influencer grows with the number of experts who follow the profiles.

3 Methods

In this section we introduce the signals and the algorithms employed for the automatic detection of influencers in Twitter by analysing their followers.

3.1 Signals

Our research is based on the work of (Rodríguez-Vidal et al., 2019), who used the textual content of the user's posts as a signal for predicting whether or not the user is an influencers. The authors show that the text in the user's posts gives useful domain information (active users in the banking domain, for instance, will use the distinctive vocabulary of the banking domain), and also provides evidence for authority, the reason being that authorities have distinctive commonalities in the way they express their opinions or transmit information. Our hypothesis is that the language employed by the followers could also be an important factor to take into account to locate influencers. For this reason, we replicate the calculus of the Language Models used by (Rodríguez-Vidal et al., 2019), which obtain a probability distribution of words, $p'(w)$, in which words likely to be included in an author message in the domain of authors are assigned high probability values; whereas other words, including those that are very ambiguous or not domain-specific but occur in the domain of authors, receive marginalized values. This distribution of words $p'(w)$ is optimized using an Expectation Maximization procedure, in the r-th iteration, is defined as:

$$p'^{(r)}(w) = \frac{p(w|L,D) * Z(w)}{(\sum_{w' \in V} p(w'|L,D) Z(w'))} \qquad (1)$$

where V is the vocabulary $w_1, ..., w_{|V|}$; L and D being the background and the target domain, respectively; $Z(w)$ is the Expectation-Step and is defined as:

$$Z(w) = \frac{(1-\lambda) p'^{(r-1)}(w)}{((1-\lambda) p'^{(r-1)}(w) + \lambda p(w))} \qquad (2)$$

$p(w|L)$ and $p(w|D)$ are defined as follows:

$$p(w|L) = \frac{tf(w,L)}{\sum_{w' \in L}(tf(w'))} \qquad (3)$$

$$p(w|D) = \frac{tf(w,D)}{\sum_{w' \in D}(tf(w'))} \qquad (4)$$

The probability of an author a belonging to the language model D is finally computed as:

$$p(D|a) = \sum_{w} (p(D|w) * p(w|a)) \qquad (5)$$

where

$$p(D|w) = Z(w)$$

$$p(w|a) \propto tf(w,Y)$$

being Y the set of tweets of the author a. After that, we compare the language of each follower with the language models of authorities (authority model) or with the language models of tweets belonging to the domain (domain model). Next, we extract from them some signals in order to explore the role played by followers in the detection of influencers. One of our main goals is to compare the utility of these signals with those extracted from the main's profiles in the work of (Rodríguez-Vidal et al., 2019).

3.1.1 Authority signals

These signals are extracted by comparing the discourse of the authorities modeled as explained in (Rodríguez-Vidal et al., 2019) with the language used by the each follower to determine the connections made by the main profile. The name, description and formula of these signals, are described below:

1. **Auth:** probability of being an authority. In order to compute this signal, we first obtain the language model for the set of followers and the language model of the authorities. We denote this signal as $P_{auth}(f)$, where $f \in F$ and F is the set of followers of a given profile.

2. **Not_Auth:** probability of being non-authority. The signal value is computed by comparing the language models for the set of followers and the language model of the non-authorities. Note that, due to the fact that the language used by the authorities and the non-authorities may contain common words, the probability of being authority is not the complementary of the non-authority probability. We denote this signal as $P_{\neg auth}(f)$, where $f \in F$ and F is the set of followers of a given profile.

3. **#_Foll_Auth:** number of followers being authorities. This signal indicates the quality of the connections made for the main profiles. It is expected that connections with the right people lead to a high probability of being influencer. In order to compute this signal, we have to estimate the probabilities of being authority and non-authority for each follower. We count the followers as influencers if they fulfil the following condition: $P_{auth}(f) - P_{\neg auth}(f) > 0$, where $f \in F$ and F is the set of followers of a given profile.

4. **Mod_Foll_Auth:** similar to the previous one, it computes if a main profile is well connected. In other words, if a main profile is followed by a high number of influencers. Like the previous signal, we have to compute, previously, the probability of being authority and not being authority. The final signal is calculated as: $\sum_{f \in F} P_{auth}(f) - P_{\neg auth}(f)$, where F is the set of followers of a given profile.

5. **Avg_Mod_Foll_Auth:** it calculates, on average, the degree of authority of the followers of each main profile. This signal is computed as $\frac{Mod\_Foll\_Auth}{num\_foll}$, where *num_foll* is the number of followers of the main profile.

6. **Prop_Foll_Auth:** ratio of followers being authorities. This signal is computed as $\frac{\#\_Foll\_Auth}{num\_foll}$, where *num_foll* is the number of followers of the main profile.

7. **Avg_Prob_Auth:** is the ratio of followers' influence. Higher values indicate that the main profile messages are validated and disseminated by several experts. This signal is calculated as: $\frac{P_{auth}(f)}{num\_foll}$, where *num_foll* is the number of followers of the main profile and f are the followers of a given profile.

8. **Sum_Foll_Auth:** is the sum of the probabilities of the followers of a main profile being influencers. This signal is calculated as $\sum_{f \in F} P_{auth}(f)$ where $f \in F$ and F is the set of followers of a given profile.

3.1.2 Domain signals

These signals are extracted by comparing the texts published in each domain (automotive and banking) with the language used by each follower to know if they possess some background knowledge about the domains. The name, description and formula of these signals, are described below:

1. **Dom:** it measures how well the discourse of the followers fits in a domain. In order to compute this signal, we first model the language of the follower set and the domains. We denote this signal as $P_{dom}(f)$, where $f \in F$ and F is the set of followers of a given profile.

2. **#_Foll_Dom:** number of followers, of a main profile, that belong to a domain. A follower f fits in a domain if she fulfils the following requirement: $P_{dom}(f) - P_{\neg dom}(f) > 0$. Note that $P_{\neg dom}(f)$ is the probability of not belonging to the domain. Due to that words can belong to different domains, the probability of belonging to a domain may not be the complementary to the probability of not belonging to it.

3. **Mod_Foll_Dom:** it computes the connections of a main profile inside a domain, in other words, whether or not a main profile is followed by people with some knowledge about a domain.

It is calculated as: $\sum_{f \in F} P_{dom}(f) - P_{\neg dom}(f)$, where $f \in F$ and F is the set of followers of a given profile.

4. **Avg_Mod_Foll_Dom:** it calculates, on average, the knowledge about a domain that the followers of a main profile have in other words, whether or not a profile is followed by experts in a domain. This signal is computed as $\frac{Mod\_Foll\_Dom}{num\_foll}$, where *num_foll* is the number of followers of given profile.

5. **Prop_Foll_Dom:** is the ratio of followers which belong to a certain domain. This signal is computed as $\frac{\#\_Foll\_Dom}{num\_foll}$, where *num_foll* is the number of followers of the main profile.

6. **Avg_Prob_Dom:** is the ratio of followers which belong to a certain domain. Higher values indicate that the posts published by the main profile can be viewed and confirmed by domain experts. This signal is calculated as $\frac{P_{dom}(f)}{num\_foll}$, where *num_foll* is the number of followers of the main profile and f are the followers of a given profile.

7. **Sum_Foll_Dom:** is the sum of the probabilities of the followers of a main profile of belonging to a domain. This signal is calculated as $\sum_{f \in F} P_{dom}(f)$ where $f \in F$ and F is the set of followers of a given profile.

3.2 Algorithms

The detection and characterization of influencers is covered as a ranking problem because it is the most natural way of presenting results to the reputation experts (RepLab 2014). We have compared the two approaches that obtained the best results for the identification of influencers in (Rodríguez-Vidal et al., 2019) to generate a ranking of users' profiles:

- **Direct Signal Rank Strategy (DSR):** each extracted signal (see section 3.1) generates a ranking of users. For instance, we can rank users by the number of followers being authorities that main profiles have. When we use two or more signals to produce a single rank, we apply a Borda voting step (Saari, 1999) to combine the ranks produced by each individual signal. If we have n elements to rank, the Borda voting lies in an ordination of the elements to consider for each signal individually in descending order assigning the higher value, in our case n, to the first element of the ranking, the $n-1$ value to the second element and so on. The combined ranking is produced by adding the values assigned to each element by every rank, and using this number to produce the final ranking.

- **Learning to Rank Strategy (L2R):** since we have training and test data, we take advantage of them and use a *machine learning* algorithm called *Learning to Rank* (Liu, 2009), to make more accurate rankings. The models created try to optimize the selected metric on the training data (in our case Mean Average Precision (MAP)). We have used MART (*Multiple Additive Regression Trees*)(Friedman, 2001) to generate rankings.

  Each main profile that L2R receives is represented as a 1-hot vector, whose length is the total number of followers that exist for all main profiles without repetition. Those positions corresponding to a real follower of a profile, is filled with its respective value (probability of being authority or belonging to the domain), the other positions contain 0 as a value, which indicates that is not a follower of the profile. To combine these vectors, we only concatenate them.

  This experimentation was carried out using the RankLib tool (Dang, 2012).

3.3 Experimental framework

One of our primary objectives is to determine how our method, which is based on signals extracted from posts published by the followers of a Twitter user, behaves for identifying influencers, and compare it with other approaches that only use information from the main profile. To do so, our experiments are performed using as main profiles the ones in the RepLab 2014 dataset. As we mentioned in the previous section, we select some signals from the followers' posts to estimate whether a user is an influencer or not.

3.3.1 Dataset

We follow the guidelines of the RepLab 2014 competition and use the Author Ranking dataset in our experiments (Amigó et al., 2014). In this task, systems are expected to "find out which authors have more reputational influence" for a given domain (automotive and banking). The required systems' output is a ranking of Twitter profiles according to their probability of being influencers with respect to the domain. The RepLab 2014 dataset consists of 7,662 Twitter profiles (all with at least 1,000 followers) related to automotive and banking domains. The profiles are divided in: 2,502 training profiles and 4,862 test profiles. Each profile contains: (i) author name; (ii) profile URL and (iii) 600 tweets published by the author (in English and Spanish). Reputational experts manually assessed each profile as influencer or not.

Since the RepLab 2014 Dataset does not supply information related to the followers of its profiles (beyond the number of followers they have), we had to collect the names and tweets of the followers of each profile in the RepLab dataset. The followers retrieved are those whose profiles were created by the extraction time of RepLab (1st June 2012-31st December 2012) and have some post published during that period of time. We gathered 600 tweets per follower, but due to the time elapsed since the dataset was built, some tweets have been lost causing some of the followers to have less than 600 tweets. Despite the main profiles were manually assessed (as influencers or not influencers) by reputation experts, we do not have such information for the followers' profiles.

This extraction was carried out using GetOldTweets-java tool (Jefferson-Henrique, 2016).

3.3.2 Metrics

Mean Average Precision (MAP) (Manning, Raghavan, y Schütze, 2008) is the official metric for RepLab 2014 competition so that, in order to compare us with the state-of-the-art systems, we have also used it. This metric measures the average precision obtained for the top k documents after each relevant document is retrieved, then this value is averaged over the information needs. This metric is mathematically expressed in Eq.6:

$$\mathrm{MAP}(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j}\mathrm{Precision}(R_{jk}) \tag{6}$$

where: m_j is the number of relevant documents at position j in the ranking and R_{jk} are the retrieved documents from the top of the ranking until the document k is reached.

3.3.3 Baselines

As reference, we have considered one naive baseline and three state-of-the-art results:

1. **Followers:** The ranking of authors according to their descending number of followers is the baseline of the RepLab 2014 competition. The number of followers is a basic indication of the author's authority potential.

2. **AleAhmad et al.** (AleAhmad et al., 2014). The main idea in this study is that influencers or opinion makers talk more about hot topics. This method extracts hot topics from each domain and a time-sensitive voting algorithm is used to rank each author on their respective topic.

3. **Rodríguez-Vidal et al.** (Rodríguez-Vidal et al., 2019). This approach got the best results in the RepLab 2014 competition. Here the authors applied language models of authorities and domain (automotive and banking) knowledge for identify and characterize influencers.

4 Results and discussion

Table 1 summarizes the results of all experiments for each signal explained in Section 3.1.1. Note that, for those experiments that only use a single signal, their results are presented for the *DSR* strategy only, since *L2R* behaves like *DSR* with one signal.

Regarding the results shown in Table 1, we can extract the following conclusions:

- The language used by the followers allows to characterize a profile much better than using only the number of followers provided by the Social Network (0.75 vs 0.38, an improvement of 97%).

- Regarding the results extracted from each signal individually, the best result

	DSR	L2R
#_Foll_Auth	0.39	-
Mod_Foll_Auth	0.44	-
Avg_Mod_Foll_Auth	0.47	-
Prop_Foll_Auth	0.44	-
Avg_Prob_Auth	0.41	-
Sum_Foll_Auth	0.42	-
#_Foll_Dom	0.35	-
Mod_Foll_Dom	0.41	-
Avg_Mod_Foll_Dom	0.42	-
Prop_Foll_Dom	0.44	-
Avg_Prob_Dom	0.42	-
Sum_Foll_Dom	0.45	-
All Combined	0.58	**0.61**
Followers	0.38	

Table 1: Signals from followers in isolation

in our experiments comes from the average authority of followers. This means that the number of followers is not as important as their average quality. This is relevant because the diffusion of a message through followers will be faster, and therefore will have a greater impact, if other influencers validate and spread that message.

- Nevertheless, combine all authority and domain signals, is the best way to use the followers' information.

Table 2 summarizes the results of combining the information extracted from the main profiles themselves with the information extracted from her followers, and compares these results against the information extracted exclusively from the followers and against other state-of-the-art systems.

	DSR	L2R
Main_Text + Followers_Text	0.74	**0.75**
Followers_Text	0.58	0.61
Rodríguez-Vidal et al.	0.68	0.74
AleAhmad et al.	0.57	
Followers	0.38	

Table 2: Adding followers' signals to the main profile

From the results shown in Table 2, we may conclude:

- The combination of the information provided by the main profile with that of her followers is a better option to locate influencers than using the information of the followers in isolation. We obtain an improvement of 18.66% for the L2R technique.
- The difference existing between the ranking techniques is irrelevant (1%), so the unsupervised approach (DSR) provides similar results than the supervised one (L2R).
- The addition of followers' signals improves overall results by a short margin (1.35%), which may indicate that the information of both actors is redundant.
- The short margin of improvement obtained and the effort taken for collecting and processing the information of the followers, lead us to think in the need to value the cost/benefit of using this approach according to the characteristics of the Social Network of study.

5 *Conclusions*

Our main goal in this study was to investigate the role of the followers in the task of finding Twitter influencers for a given domain. To do so, we model their language and extract different signals which help us to characterize authorities and domain experts and we compare our results against the state-of-the-art. The main conclusions of our experiments are:

- The followers of a user may provide useful information for characterizing her. The profiles followed by other influencers are more likely to be influencers. This indicates that influencers tend to be connected with other of their kind.
- This discovery leads us to an interesting discussion. Since influencers tend to be connected to each other, an idea written by one of them is accepted and validated (e.g. using a retweet) by other influencers, and this may cause more severe reputational crises since: (i) the diffusion of a message by other influencers adds a new audience to that idea; and (ii) if the opinions of one influencer are reliable for the users of a given community, the validation by another influencer(s) gives the message a greater veracity in the eyes of that community.
- The combination of the information provided by the main profiles with the information of her followers produces the

best results, but only by a short margin, which may indicate that these information is redundant.

Acknowledgments

This research was supported by the Spanish Ministry of Science and Innovation (Vemodalen Project, TIN2015-71785-R).

Bibliografía

AleAhmad, A., P. Karisani, M. Rahgozar, y F. Oroumchian. 2014. University of tehran at replab 2014. En *Conference and Labs of the Evaluation Forum (Working Notes)*, páginas 1528–1536.

Amigó, E., J. Carrillo-de Albornoz, I. Chugur, A. Corujo, J. Gonzalo, E. Meij, M. de Rijke, y D. Spina. 2014. Overview of replab 2014: author profiling and reputation dimensions for online reputation management. En *International Conference of the Cross-Language Evaluation Forum for European Languages*, páginas 307–322. Springer.

Aral, S. y D. Walker. 2012. Identifying influential and susceptible members of social networks. *Science*, 337(6092):337–341.

CLEF. 2014. Replab 2014 campaign @online. http://nlp.uned.es/replab2014/.

Cossu, J.-V., K. Janod, E. Ferreira, J. Gaillard, y M. El-Bèze. 2014. Lia@ replab 2014: 10 methods for 3 tasks. En *4th International Conference of the Conference and Labs of the Evaluation Forum initiative*. Citeseer.

Cossu, J.-V., V. Labatut, y N. Dugué. 2016. A review of features for the discrimination of twitter users: application to the prediction of offline influence. *Social Network Analysis and Mining*, 6(1):1–23.

Dang, V. 2012. The lemur project-wiki-ranklib. *Lemur Project,[Online]. Available:* `http://sourceforge.net/p/lemur/wiki/RankLib`.

Friedman, J. H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, páginas 1189–1232.

Jefferson-Henrique. 2016. Getoldtweets-java @online. https://github.com/Jefferson-Henrique/GetOldTweets-java, Abril.

Liu, T.-Y. 2009. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3(3):225–331, Marzo.

Lomena, J. J. M. y F. L. Ostenero. 2014. Uned at clef replab 2014: Author profiling task. En *Conference and Labs of the Evaluation Forum (Working Notes)*, páginas 1537–1546.

Madden, M. y A. Smith. 2010. Reputation management and social media. *pewresearch.org,[Online]. Available:* `pewresearch.org/internet/2010/05/26/reputation-management-and-social-media/`.

Manning, C. D., P. Raghavan, y H. Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, USA.

Nebot, V., F. Rangel, R. Berlanga, y P. Rosso. 2018. Identifying and classifying influencers in twitter only with textual information. En *International Conference on Applications of Natural Language to Information Systems*, páginas 28–39. Springer.

Rodríguez-Vidal, J., J. Gonzalo, L. Plaza, y H. A. Sánchez. 2019. Automatic detection of influencers in social networks: Authority versus domain signals. *Journal of the Association for Information Science and Technology.*

Saari, D. G. 1999. Explaining all three-alternative voting outcomes. *Journal of Economic Theory*, 87:313–355.

Vilares, D., M. Hermo, M. A. Alonso, C. Gómez-Rodríguez, y J. Vilares. 2014. Lys at clef replab 2014: Creating the state of the art in author influence ranking and reputation classification on twitter. En *Conference and Labs of the Evaluation Forum (Working Notes)*, páginas 1468–1478.

Villatoro-Tello, E., G. Ramírez-de-la Rosa, C. Sánchez-Sánchez, H. Jiménez-Salazar, W. A. Luna-Ramírez, y C. Rodríguez-Lucatero. 2014. Uamclyr at replab 2014: Author profiling task. En *Conference and Labs of the Evaluation Forum (Working Notes)*, páginas 1547–1558.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 29-36 recibido 08-11-2019 revisado 15-01-2020 aceptado 18-01-2020

Tarántula –> araña –> animal: asignación de hiperónimos de segundo nivel basada en métodos de similitud distribucional

Tarantula –> spider –> animal: second level hypernymy discovery based on distributional similarity methods

Rogelio Nazar, Javier Obreque, Irene Renau
Instituto de Literatura y Ciencias del Lenguaje
Pontifica Universidad Católica de Valparaíso
rogelio.nazar@pucv.cl, j.obrequezamora@gmail.com, irene.renau@pucv.cl

Resumen: La asignación automática de hiperónimos sigue presentando problemas para el procesamiento del lenguaje natural. En particular, los sustantivos polisémicos se vinculan a distintos hiperónimos y por ello pueden causar problemas estructurales en una taxonomía léxica. Por ejemplo, el sustantivo *tarántula* puede ser registrado como hipónimo de *araña* y, como este es un sustantivo polisémico (puede denotar a un ser vivo o a un tipo de lámpara), es necesario determinar cuál es el hiperónimo siguiente en la cadena: *animal* o *artefacto*. En el presente artículo exploramos métodos para resolver este problema utilizando el cálculo de la similitud entre sustantivos utilizando como variable predictora los verbos con los que coocurren. Los mejores resultados (84 % de acierto) se obtienen con un método simple que solo mide coocurrencia, sin tener en cuenta información sintáctica.
Palabras clave: hiperonimia, polisemia, similitud distribucional, taxonomía

Abstract: Automatic hypernymy discovery continues to present challenges for natural language processing. Polysemous nouns are linked to more than one hypernym and can therefore cause structural damage on a lexical taxonomy. For instance, the Spanish noun *tarántula* ('tarantula') is a hyponym of *araña* ('spider'), but this is also a polysemous noun, as it means 'chandelier' as well. It is thus necessary to determine the next hypernym in the chain, that is *animal* ('animal') or *artefacto* ('artifact'). In this paper we explore methods to solve this problem using a similarity measure that uses verb-noun co-occurrence as a predictor variable. Best results (84 % success) are obtained with a simple method that only measures co-occurrence, irrespective of any syntactic information.
Keywords: distributional similarity, hypernymy, polysemy, taxonomy

1 Introducción

El establecimiento de relaciones de hiperonimia entre unidades léxicas continúa siendo un desafío en el campo del procesamiento del lenguaje natural. Actualmente, las estrategias que existen alcanzan promedio de precisión que fluctúa en torno al 80 % (Velardi, Faralli, y Navigli, 2013; Bordea, Lefever, y Buitelaar, 2016), lo cual deja un amplio margen de mejora.

En el marco de la inducción automática de taxonomías léxicas, es decir, las estructuras que emergen de las relaciones de hiperonimia-hiponimia (Lyons, 1977), uno de los problemas pendientes es cómo tratar adecuadamente el fenómeno de la polisemia (Bordea et al., 2015; Klapaftis y Manandhar, 2010). Eso es así, al menos, en el caso de las taxonomías semasiológicas, es decir, aquellas que se basan en unidades léxicas, como es usual en lexicografía, y no en conceptos, que es lo propio de las ontologías (Baldinger, 1977; Sager, 1990).

La polisemia es el fenómeno por el cual una palabra tiene más de un significado, y ocurre cuando uno de los significados da origen a otro u otros, por metáfora, metonimia u otro mecanismo, sin que el significado original se anule (Ullmann, 1972; Lyons, 1977; Kilgarriff, 1992; De Miguel, 2016). Así, de *araña* 'animal' se deriva *araña* 'lámpara' por la similitud formal entre ambas entidades.

En este trabajo se aborda específicamente el problema de la herencia semántica en la

ISSN 1135-5948. DOI 10.26342/2020-64-3

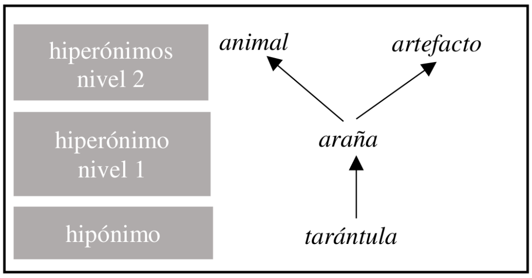


Figura 1: Estructura taxonómica con un hiperónimo de nivel 1 y 2 de nivel 2.

cadena de relaciones de hiperónimo-hipónimo en una taxonomía léxica. Con el fin de ilustrar la problemática, imaginemos el caso de un algoritmo de asignación de hiperónimos que establece correctamente la relación de hiponimia del sustantivo *tarántula* con respecto a *araña*. Tal como se muestra en la Figura 1, sería necesario entonces determinar de qué significado específico del sustantivo *araña* se trata, ya sea el de *animal* (solución correcta) o el de *artefacto* (solución incorrecta). En este trabajo, se llamará *hiperónimo de nivel* 1 al hiperónimo del tipo *araña* (inmediatamente superior en la cadena hiperonímica al hipónimo, en este caso *tarántula*), e hiperónimo de nivel 2 al hiperónimo del tipo *animal* o *artefacto* (de los cuales solo uno de ellos es correcto, en este caso *animal*).

El objetivo de esta investigación es, entonces, proponer un algoritmo para resolver los casos de ambigüedad de hiperónimos de nivel 2. El método está basado en medidas de coocurrencia léxica, a través de las cuales es posible seleccionar el significado correcto entre los que ofrece un hiperónimo polisémico. Para ello, en esta investigación empleamos la coocurrencia sustantivo-verbo. Así, los verbos con los que frecuentemente coocurre el sustantivo *tarántula* proporcionarán pistas sobre si se debe clasificar como *animal* o como *artefacto*. Consideramos que la conformación de este método representa un avance en el marco de la inducción automática de taxonomías y puede contribuir a solucionar el problema de la polisemia en hiperónimos de segundo nivel.

A continuación, se presenta un breve estado de la cuestión (apartado 2), la metodología (apartado 3), los resultados y evaluación (apartado 4) y las conclusiones y trabajo futuro (apartado 5). El código fuente del proyecto, implementado en el lenguaje Perl, se encuentra disponible en la página web que acompaña el artículo: http://www.tecling.com/hat

2 La asignación automática de hiperónimos

La hiperonimia es una de las relaciones semánticas de inclusión que acontecen en la estructura léxica de una lengua (García y Pascual, 2009). Leech (1985) la describió como el fenómeno por el cual una palabra incluye semánticamente a otra. Así, un hiperónimo se define como una unidad léxica cuyo significado está en un nivel de abstracción más alto que el de su hipónimo.

Una taxonomía léxica debe presentar las siguientes tres características fundamentales:

1. **Herencia**: un nodo inferior (hipónimo) hereda las propiedades de su nodo superior (hiperónimo).
2. **Asimetría**: una unidad léxica no puede ser superior (hiperónimo) e inferior (hipónimo) de otra unidad léxica al mismo tiempo.
3. **Transitividad**: si un hipónimo a tiene un hiperónimo directo ($a \rightarrow b$) y este, a su vez, tiene otro ($b \rightarrow c$), entonces el primero es hipónimo del último ($a \rightarrow c$).

En lingüística computacional se utilizaron términos como *red semántica* u *ontología* para referirse a estructuras de datos relacionados formalmente aplicadas al procesamiento automático de grandes cantidades de datos (Sowa, 2000). Cabe aclarar, sin embargo, que una taxonomía léxica es algo distinto a las anteriores estructuras, ya que solo establece relaciones de hiperonimia y además lo hace entre unidades léxicas, no entre conceptos. Una ontología no debería presentar problemas derivados de la polisemia porque puede identificar sus nodos conceptuales con un código arbitrario, tal como un identificador numérico. Esto le permite, además, asociar términos distintos para un mismo concepto, con lo cual se evita también el problema de la sinonimia, otra de las complicaciones de las taxonomías semasiológicas.

Los primeros intentos para construir taxonomías y ontologías se desarrollaron de forma manual, en casos como los de CyC (Lenat, 1995), WordNet (Fellbaum, 1998), EuroWordNet (Vossen, 2004), Snomed (Stearns et al., 2001), entre otros. Por supuesto, el desarrollo en forma manual de estas estructuras de datos presenta limitaciones. Por un lado, son propensas a inconsistencias, incluso con protocolos rigurosos. Por otro lado, se vuelven obsoletas con relativa rapidez debido al dinamismo de la lengua, problema que se agudiza en el caso de las taxonomías especializadas, que tienen una acelerada evolución terminológica.

Estas limitaciones han sido motivo suficiente para emprender la tarea de la generación automática de taxonomías. Una primera línea de investigación consistió en la extracción de relaciones de hiperonimia a través del procesamiento automático de diccionarios (Calzolari, 1984; Chodorow, Byrd, y Heidorn, 1985; Guthrie et al., 1990; Agirre et al., 1994). Estos trabajos se basan fundamentalmente en la elaboración de sistemas de reglas que puedan analizar las definiciones y extraer pares hipónimo-hiperónimo. Una regla de este tipo puede ser que el primer sustantivo en la definición de un sustantivo será su hiperónimo.

Más tarde se aplicó una idea similar, es decir, la utilización de listas de patrones preestablecidos, no ya sobre diccionarios sino sobre texto libre (Hearst, 1992). Se realizaron múltiples variantes de este enfoque, tales como el intento de extraer estos patrones directamente del mismo corpus de manera inductiva (Snow, Jurafsky, y Ng, 2006).

En la actualidad, la inducción de taxonomías sigue enfrentando, entre otros, problemas de estructura y polisemia. En este marco, el presente estudio contribuye a mejorar los resultados de la inducción de taxonomías mediante una propuesta metodológica que se fundamenta en dos ámbitos: por una parte, en los principios de la semántica léxica, utilizando unidades léxicas de contextos sintagmáticos de los sustantivos en estudio; por otra parte, en la estadística de corpus fundada, en nuestro caso, en la aplicación de una medida de similitud distribucional (Grefenstette, 1994; Lin, 1998) que permitirá operacionalizar la similitud semántica entre palabras con el fin de obtener mediciones cuantificables y comprobables empíricamente.

3 Metodología

En esta sección detallamos la propuesta metodológica para asignar relaciones de hiperonimia de segundo nivel en los casos de polisemia, utilizando para ello una medida de similitud distribucional entre sustantivos. Como variable para la comparación, utilizamos los verbos con los que están asociados sintagmáticamente los sustantivos. Presentamos primero la versión más básica del método y luego una serie de variantes que van añadiendo complejidad.

La sección abre con la descripción del proceso de selección de la muestra para experimentación (3.1). Luego se presenta el método más básico, llamado *binario*, que utiliza vectores binarios y solo mide la frecuencia de coocurrencia entre verbos y sustantivos (3.2). A continuación, se describe el resto de las variantes del método: *ponderado*, que también utiliza vectores binarios pero con verbos seleccionados mediante una medida de asociación (3.3); *euclidiano*, que en lugar de vectores binarios utiliza números reales obtenidos a partir de la medida de asociación (3.4) y, finalmente, *dependencias*, que utiliza vectores binarios pero con verbos que se obtienen por relaciones de dependencia sintáctica (3.5).

3.1 Selección de la muestra para experimentación

Con el objeto de obtener tríadas como las mostradas en la Figura 1 (es decir, hipónimo + hiperónimo de nivel 1 + dos posibles hiperónimos de nivel 2), implementamos un script Perl que interroga la base de datos WordNet en castellano (Vossen, 2004). El script detecta la presencia de sustantivos en más de un synset, lo que puede ser interpretado como indicador de polisemia, y que muestren a la vez por lo menos un hipónimo. Esto permitió encontrar 26 casos que satisficieran el requerimiento de una frecuencia mínima de 100 ocurrencias en el corpus de trabajo (v. apartado 3.2.1). Como se puede ver en la Tabla 1, en cada tríada tenemos el hipónimo, que es el sustantivo objetivo (como *laucha*), el sustantivo polisémico es el hiperónimo de primer nivel (*ratón*) y el resto son los candidatos a hiperónimo de nivel 2 (*animal* o *artefacto*), entre los cuales el sistema debe elegir uno.

3.2 El método base: *binario*

3.2.1 Extracción de contextos de aparición de los sustantivos

Por cada sustantivo analizado tomamos una muestra de contextos de aparición. Para ello utilizamos el corpus esTenTen (Kilgarriff y Renau, 2013), versión 2011 (9.500 millones de palabras). Hicimos muestreos aleatorios de un máximo de hasta 5.000 concordancias por

Sust. objetivo	Hiper. nivel 1	Hiper. nivel 2
asaltante	ladrón	**humano** — artefacto
caniche	perro	**animal** — artefacto
chimpancé	mono	**animal** — prenda
laucha	ratón	**animal** — artefacto
tarántula	araña	**animal** — artefacto
...	...	...

Tabla 1: Ejemplos del tipo de tríada en estudio, con la opción correcta en negrita.

cada sustantivo, aunque en muchos casos la muestra fue menor debido a que no todos tienen tanta frecuencia de aparición en el corpus. Utilizamos una ventana de contexto de 10 palabras a derecha e izquierda, teniendo en cuenta la distancia variable en que se puede presentar la coocurrencia verbo-argumento y, en esta variante del método, nos limitamos a medir la frecuencia de coocurrencia. Para ello basta el etiquetado morfosintáctico del corpus EsTenTen. La Tabla 2 muestra un fragmento de contexto del sustantivo *tarántula* con el etiquetado del corpus.

Forma	Categoría gramatical	Lema
Si	CSUBX	si
una	ART	un
tarántula	NC	tarántula
pica	VLfin	picar
a	PREP	a
una	ART	un
persona	NC	persona
...	...	...

Tabla 2: Ejemplo de contexto de aparición del sustantivo *tarántula.*

3.2.2 Extracción de los verbos

Por cada uno de los sustantivos analizados, se recorrieron sus contextos de aparición registrando la frecuencia de los verbos con los que coocurren. De este modo, conservamos los verbos en los que se observa una frecuencia de coocurrencia de mínimo 5 casos, umbral arbitrario sobre el que es más improbable que la observación sea fruto de accidente o error.

3.2.3 Conformación de una matriz de coocurrencia con verbos

Una vez obtenidos los listados de coocurrencia sustantivo-verbo, se conformó una matriz $M_{i,j}$ en la que los sustantivos son dispuestos en las filas y los verbos con los que coocurren en las columnas. La Tabla 3 muestra la estructura de esta matriz. Uno de los sustantivos analizados, como *absolutismo*, coocurre con frecuencia con el verbo *abandonar*, lo que también sucede con el sustantivo *caniche* pero no así con *tarántula.*

	abandonar	*abarcar*	*abogar*	...
absolutismo	1	0	0	...
caniche	1	0	0	...
tarántula	0	0	0	...
...	...	...	...	...

Tabla 3: Ejemplificación de la matriz de coocurrencia sustantivo-verbo.

En esta variante del método optamos por valores binarios, como se muestra en la Tabla 3. El valor de la celda se define en (1), donde la frecuencia de coocurrencia $fr(i,j)$ debe superar un umbral u ($u = 5$).

$$M_{i,j} = \begin{cases} 1 & fr(i,j) > u) \\ 0 & \text{otherwise} \end{cases} \qquad (1)$$

3.2.4 Formación de vectores clase para hiperónimos de segundo nivel

Los hiperónimos de segundo nivel utilizados en la muestra (por ejemplo *evento, animal, máquina*, etc.) resultan demasiado abstractos para crear un vector de coocurrencia directamente como se explica en el apartado 3.2.3. Esto motivó que la construcción de vectores se llevara a cabo de manera indirecta, a través de la suma de vectores de varios sustantivos pertenecientes a esas categorías. La Tabla 4 muestra algunos ejemplos de dicha selección, donde se ve la categoría y 10 sustantivos pertenecientes a ella. La selección de sustantivos es arbitraria, pero se trata en todos los casos de miembros prototípicos de cada categoría, y que tendrán frecuencia alta en el corpus.

Hiper. nivel 2	Hipónimos
animal	caballo, canario, canguro, delfín, elefante, gorrión, jirafa, león, lobo, ornitorrinco
máquina	aspiradora, automóvil, cocina, cortadora, estufa, horno, juguera, motocicleta, refrigerador, soldadora
prenda	blusa, calcetín, calzón, camisa, chaleco, cinturón, corbata, pantalón, polera, sudadera
...	...

Tabla 4: Ejemplo de construcción de vectoresclase.

Por cada sustantivo de estas categorías se extrajeron sus contextos de aparición tal como se describe en (3.2.1) y se construyó una matriz de coccurrencia sustantivo-verbo como en (3.2.3). La Tabla 5 muestra la forma en que se suman los vectores-miembro para obtener un vector-clase. Al igual que en la Tabla 4, los sustantivos se disponen en las filas y los verbos en las columnas. La diferencia aquí está en que estos sustantivos (H_i) son los diez miembros elegidos de cada categoría. La última fila, señalada con el símbolo VC, representa el vector-clase, y consiste en la suma de los vectores de cada uno de sus hipónimos. Esto significa que cada componente de VC tendrá valor 1 si existe al menos una celda con valor 1 en la columna correspondiente. De esta manera, por cada uno de esos tipos semánticos más abstractos, obtenemos un vector clase, representado por los verbos con los que coocurren sustantivos hipónimos de estos hiperónimos más abstractos.

	V_1	V_2	V_3	V_4	V_5	...	V_n
H_1	0	1	1	0	0	...	0
H_2	0	0	1	0	1	...	0
H_3	0	1	1	0	0	...	1
H_4	0	0	1	0	1	...	0
...	...	...	...	...	...	...	...
H_n	0	1	1	0	0	...	0
VC	0	1	1	0	1	...	1

Tabla 5: Esquematización de la suma de vectores para la conformación del vector-clase.

3.2.5 Cálculo de similitud entre vectores

Una vez poblada la matriz de los sustantivos objetivo (Tabla 3) y la de los vectores-clase (Tabla 5), el siguiente paso consiste en aplicar una medida de similitud entre el vector que corresponde a este sustantivo objetivo ($\vec{o}$) y cada uno de los vectores-clase ($\vec{VC}$) que representan a los hiperónimos de segundo nivel. Como medida de similitud aplicamos el índice de Jaccard (2), que es apropiado para la comparación de vectores binarios. Dados dos vectores A y B, la similitud se obtiene oponiendo la intersección a la unión.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

Así, en el caso de *tarántula*, la selección entre *animal* y *artefacto* (sus dos hiperónimos de segundo nivel) se realiza por medio de una función h según el valor de similitud recién explicado, tal como se muestra en la Ecuación 3, donde $\vec{o}$ puede ser *tarántula*, $\vec{VC_k}$ puede ser *animal* y $\vec{VC_i}$ *artefecto*. Siempre se elige una de las opciones.

$$h(\vec{o}) = \begin{cases} \vec{VC_k} & J(\vec{o}, \vec{VC_k}) > J(\vec{o}, \vec{VC_i}) \\ \vec{VC_i} & \text{otherwise} \end{cases} \tag{3}$$

3.3 Aplicación de una medida de asociación: la variante *ponderada*

Tal como anticipamos al comienzo de la sección, experimentamos con distintas variantes del método principal con el fin de contrastar resultados.

La variante *ponderada* del método es muy similar a la anterior, y sigue utilizando vectores binarios. La única diferencia es que ahora poblamos esos vectores mediante una mejor selección de los verbos, utilizando para ello una medida de asociación sintagmática (Ecuación 4). De forma similar a la Ecuación 1, solo tendrán valor 1 los pares sustantivo-verbo que tengan una ponderación mayor a un umbral mínimo que, a diferencia del caso anterior, ahora tiene otro valor ($u = 0{,}01$).

$$cooc(s,v) = \frac{f(s,v)}{\sqrt{f(s)}.\sqrt{f(v)}} \tag{4}$$

El resto del procedimiento es idéntico al método básico.

3.4 Uso de vectores con números reales en lugar de binarios: la variante *euclidiana*

Esta tercera variante del método está basada en la anterior (*ponderada*), pero en lugar de utilizar valores binarios ahora son vectores de números reales, cuyos valores se obtienen de la ponderación definida en la Ecuación 4. El uso de valores reales en lugar de binarios obliga a hacer ajustes en el método básico, como la conformación de los vectores-clase (apartado 3.2.4). En este caso, los valores de los vectores-clase se obtienen sumando las ponderaciones de los verbos asociados a cada sustantivo, como se indica en la Ecuación 5.

$$VC_j = \sum_{i=1}^{|VC|} H_{i,j} \tag{5}$$

Otra de las diferencias en esta variante del método es que al utilizar vectores con números reales podemos optar por otras medidas de similitud. En este caso optamos por la utilización de la distancia euclidiana, definida en la Ecuación 6.

$$d(A,B) = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2} \qquad (6)$$

Esta variante permite captar la idea según la cual un sustantivo objetivo o y un hiperónimo de segundo nivel VC_k correcto deberían tener un similar perfil de coocurrencia con verbos, siendo algunos verbos más significativos que otros. Con esta medida se selecciona un hipónimo de la misma forma que en la Ecuación 3, solo que en este caso el signo $<$ se invierte a $>$, ya que se trata de una medida de distancia en lugar de similitud.

3.5 Uso de un *parser* sintáctico para extraer relaciones verbo-sustantivo: la variante *dependencias*

La última variante, y la más compleja, involucra la utilización de un analizador de dependencias sintácticas para determinar la función gramatical que se produce entre sustantivos y verbos. El *parser* permite limitar la selección a las parejas sustantivo-verbo que efectivamente contraen una relación sintáctica, como puede ser el caso de la relación sujeto-verbo, verbo-objeto directo, etc.

Utilizamos para ello UDPipe, uno de los mejores y más recientes analizadores de dependencias (Straka y Straková, 2017). La Tabla 6 muestra el resultado del análisis sintáctico del mismo fragmento de contexto del sustantivo *tarántula* que se mostró en la Tabla 2. En este caso se produce un error, ya que el verbo *picar* no es reconocido como tal (línea 96) y se etiqueta como adjetivo, perdiéndose así la información relativa a que *tarántula* es el sujeto del verbo *picar*. Dada la mayor complejidad de este tipo de análisis, cabe esperar que se produzca una alta tasa de error. Sin embargo, al mismo tiempo resulta razonable suponer también que la gran cantidad de contextos de aparición de los sustantivos compense esta tasa de error.

En este caso optamos nuevamente por la utilización de vectores binarios, pero no aplicamos un límite de frecuencia por ser ya esta variante muy selectiva. De este modo, si se observa al menos una vez que existe una relación sintáctica entre un sustantivo y un verbo, el valor correspondiente a esa celda será 1.

Línea	Forma	Lema	POS	Dep.	Func.
93	Si	si	SCONJ	102	mark
94	una	uno	DET	95	det
95	tarántula	tarántula	NOUN	102	nsubj
96	pica	pico	ADJ	95	amod
97	a	a	ADP	99	case
98	una	uno	DET	99	det
99	persona	persona	NOUN	95	nmod
...	...	...	...	...	...

Tabla 6: Ejemplo de análisis de dependencias con *UDPipe* en que se produce un error en la detección del verbo *picar*.

4 Resultados

La Tabla 7 muestra los resultados del método en sus distintas variantes, sobre la muestra de sustantivos objetivo. La cobertura es igual a la precisión debido a que forzamos al sistema a elegir siempre una de las opciones.

Variante	Precisión
binaria	69 %
ponderada	**84 %**
euclidiana	73 %
dependencias	57 %
aleatoria	50 %

Tabla 7: Resultados del método en sus distintas variantes

Los mejores resultados se obtuvieron con la variante *ponderada*. Atribuimos este resultado a una mejor selección de los verbos coocurrentes, a través de una medida de asociación sintagmática. Esto suprimió el ruido que introducían en la variante *binaria* los verbos que tienen alta frecuencia de coocurrencia con muchos sustantivos diferentes. La variante *euclidiana*, con la incorporación de la medida de distancia euclidiana, también mejora la variante *binaria*, pero con resultados más modestos. Finalmente, la variante *dependencias*, que extrae los verbos mediante análisis sintáctico y es la de mayor complejidad, tuvo el peor desempeño, con solo 7 puntos por encima de una clasificación aleatoria. Esto puede ser atribuible al hecho de que los textos del corpus, tomados de páginas web, contienen una sintaxis y ortografía relajadas propias de los textos de Internet. Solucionar este problema queda fuera del alcance de la presente investigación.

La Tabla 8 muestra los resultados con las primeras 9 unidades analizadas por orden alfabético en el caso de la variante *ponderada*, que fue la que produjo mejores resultados. La primera columna indica la evaluación: 1 si el ensayo es exitoso y 0 si no lo es. La si-

E	d	h_1	h_2	S
1	absolutismo	sistema	concepto	16.46
			máquina	8.47
0	acueducto	canal	institución	11.55
			lugar	8.12
0	albahaca	planta	lugar	14.15
			servivo	11.09
1	ametrallador	cañón	arma	5.56
			lugar	2.58
1	asaltante	ladrón	humano	11.46
			artefacto	8.07
1	caniche	perro	animal	6.26
			artefacto	3.84
1	cencerro	campana	instr. musical	3.07
			máquina	2.97
1	chimpancé	mono	animal	10.47
			prenda	5.75
1	dedo	miembro	parte cuerpo	11.39
			humano	10.92

Tabla 8: Ejemplos de resultados con la variante *ponderada*

guiente columna presenta el sustantivo objetivo (o), la siguiente el hiperónimo de primer nivel ($h1$) y la siguiente los distintos hiperónimos de segundo nivel ($h2$). La última columna (S) indica el valor obtenido con el índice de Jaccard entre el vector de coocurrencia del sustantivo objetivo y, en cada caso, el hiperónimo de segundo nivel (el valor más alto se presenta primero). Entre los 26 casos estudiados hay 22 exitosos, lo que representa un resultado estadísticamente significativo ($p = 0{,}0005$).

5 Conclusiones y trabajo futuro

En este trabajo hemos presentado una propuesta metodológica para desambiguar la relación entre un sustantivo y un hiperónimo polisémico en el contexto de la inducción automática de taxonomías. El método, fundado en una medida de similitud distribucional, se basa en la idea según la cual las palabras que aparecen en cotextos similares tienden a tener significados similares.

En la propuesta, se han restringido los cotextos en función de los verbos con los que coocurren los sustantivos en estudio, ya que se trata de una clase abierta de palabras, pero al mismo tiempo limitada (cerca de 6.000 verbos aparecen con cierta frecuencia en el EsTenTen). Esta característica convierte a los verbos en predictores útiles para obtener información semántica acerca de los sustantivos con los que coocurren, ya que representan una matriz más manejable que una de adjetivos o sustantivos.

Además del método básico, hemos explorado distintas variantes. Utilizamos vectores binarios que indican la coocurrencia sustantivo-verbo, y también vectores con valores reales; probamos el uso de simple frecuencia de coocurrencia y luego una medida de asociación estadística y, finalmente, hemos explorado también la posibilidad de extraer los verbos por medio de un analizador de dependencias sintácticas.

Nuestros resultados permiten concluir que la mejor variante es la que utiliza vectores binarios que miden frecuencia de coocurrencia sustantivo-verbo, seleccionando los verbos con una medida de asociación (la variante *ponderado*). Creemos que la tasa de éxito es remarcable, teniendo en cuenta que se trata de un método relativamente simple. No existen, que sepamos, propuestas similares para la adjudicación de hiperónimos de segundo nivel en casos de polisemia.

En cuanto a trabajo futuro, es necesario continuar introduciendo nuevas variantes metodológicas y reproducir los experimentos con muestras más grandes de datos, ya que esto permitiría estudiar mejor cómo afectan a los resultados las diferencias de frecuencia y de prototipicidad de cada significado, una variable que conviene controlar en un diseño de investigación de este tipo (por ejemplo, en el caso del sustantivo *perro*, el sentido de *animal* tendrá más peso que el de *artefacto*). También sería necesario probar la utilización de ventanas oracionales en lugar de ventanas de contexto simétricas. Otra posibilidad sería reproducir el mismo método pero utilizando adjetivos o incluso sustantivos en lugar de verbos. Finalmente, proyectamos reproducir los experimentos en otras lenguas (francés, inglés, etc.) en el contexto de nuestro proyecto KIND[1] de taxonomías automatizadas en varias lenguas.

Agradecimientos

Esta investigación ha sido posible gracias al financiamiento del Proyecto Fondecyt Regular 1191204 "Polisemia regular de los sustantivos del español: análisis semiautomático de corpus, caracterización y tipología", dirigido por Irene Renau. Agradecemos también a los revisores por sus útiles comentarios.

[1] http://www.tecling.com/kind

Bibliografía

Agirre, E., X. Arregi, X. Artola, A. D. de Ilarraza, y K. Sarasola. 1994. A methodology for the extraction of semantic knowledge from dictionaries using phrasal patterns. En *Proceedings of IBERAMIA'94. IV Congreso Iberoamericano de Inteligencia Artificial*, páginas 263–270, Caracas (Venezuela).

Baldinger, K. 1977. *Teoría semántica: hacia una semántica moderna.* Coleccion Romania. Alcala.

Bordea, G., P. Buitelaar, S. Faralli, y R. Navigli. 2015. SemEval-2015 Task 17: Taxonomy extraction evaluation (texeval). En *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, páginas 902–910. ACL.

Bordea, G., E. Lefever, y P. Buitelaar. 2016. SemEval-2016 Task 13: Taxonomy extraction evaluation (texeval-2). En *SemEval-2016*, páginas 1081–1091. ACL.

Calzolari, N. 1984. Detecting patterns in a lexical data base. En *Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on ACL*, páginas 170–3. ACL.

Chodorow, M. S., R. J. Byrd, y G. E. Heidorn. 1985. Extracting semantic hierarchies from a large on-line dictionary. En *Proceedings of the 23rd annual meeting on ACL*, páginas 299–304. ACL.

De Miguel, E. 2016. Lexicología. En J. Gutiérrez, editor, *Enciclopedia de Lingüística Hispánica.* Ariel, Barcelona, páginas 153–185.

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database.* Bradford Books.

García, R. y J. Pascual. 2009. Relaciones de significado entre las palabras. En E. D. Miguel, editor, *Panorama de lexicología.* Ariel, Barcelona, páginas 117–131.

Grefenstette, G. 1994. *Explorations in Automatic Thesaurus Discovery.* Kluwer Academic Publishers, Norwell, MA, USA.

Guthrie, L., B. Slator, Y. Wilks, y R. Bruce. 1990. Is there content in empty heads? En *Proc. of the 13th International Conference on Computational Linguistics, COLING'90 (Helsinki, Finland)*, páginas 138–143.

Hearst, M. A. 1992. Automatic acquisition of hyponyms from large text corpora. En *Proceedings of the 14th Conference on Computational Linguistics - Volume 2*, COLING '92, páginas 539–545, Stroudsburg, PA, USA. ACL.

Kilgarriff, A. 1992. *Polisemy.* Ph.D. tesis. University of Sussex.

Kilgarriff, A. y I. Renau. 2013. estenten, a vast web corpus of peninsular and american spanish. *Procedia - Social and Behavioral Sciences*, 95:12 – 19.

Klapaftis, I. P. y S. Manandhar. 2010. Taxonomy learning using word sense induction. En *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, páginas 82–90, Los Angeles, California, Junio. ACL.

Leech, G. 1985. *Semántica.* Alianza Universal, No. 197. Alianza.

Lenat, D. 1995. Cyc: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):33–38, Noviembre.

Lin, D. 1998. Automatic retrieval and clustering of similar words. En *Proceedings of the 17th International Conference on Computational Linguistics - Volume 2*, COLING '98, páginas 768–774, Stroudsburg, PA, USA. ACL.

Lyons, J. 1977. *Semantics*, volumen 2. Cambridge University Press.

Sager, J. C. 1990. *A Practical Course in Terminology Processing.* John Benjamins, Amsterdam/Philadelphia.

Snow, R., D. Jurafsky, y A. Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. En *Proceedings of the 21st International Conference on Computational Linguistics, Sydney, Australia, 17-21 July 2006.*

Stearns, M. Q., C. Price, K. A. Spackman, y A. Y. Wang. 2001. Snomed clinical terms: overview of the development process and project status. En *Proceedings of the AMIA Symposium*, páginas 662–666. American Medical Informatics Association.

Straka, M. y J. Straková. 2017. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. En *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, páginas 88–99, Vancouver, Canada, Agosto. ACL.

Ullmann, S. 1972. *Semántica.* Aguilar.

Velardi, P., S. Faralli, y R. Navigli. 2013. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707.

Vossen, P. 2004. Eurowordnet: A multilingual database of autonomous and language-specific wordnets connected via an inter-lingual-index. *Special Issue on Multilingual Databases, International Journal of Linguistics*, 17(2):161–173, 06.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 37-44 recibido 18-11-2019 revisado 10-01-2020 aceptado 12-01-2020

Predicting the Humorousness of Tweets Using Gaussian Process Preference Learning

Identificando el humor de tuits utilizando el aprendizaje de preferencias basado en procesos gaussianos

Tristan Miller[1], Erik-Lân Do Dinh[2], Edwin Simpson[2], Iryna Gurevych[2]
[1]Austrian Research Institute for Artificial Intelligence (OFAI)
Freyung 6, 1010 Vienna, Austria
tristan.miller@ofai.at
[2]Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Department of Computer Science, Technische Universität Darmstadt
Hochschulstraße 10, 64289 Darmstadt, Germany
https://www.ukp.tu-darmstadt.de/

Abstract: Most humour processing systems to date make at best discrete, coarse-grained distinctions between the comical and the conventional, yet such notions are better conceptualized as a broad spectrum. In this paper, we present a probabilistic approach, a variant of Gaussian process preference learning (GPPL), that learns to rank and rate the humorousness of short texts by exploiting human preference judgments and automatically sourced linguistic annotations. We apply our system, which is similar to one that had previously shown good performance on English-language one-liners annotated with pairwise humorousness annotations, to the Spanish-language data set of the HAHA@IberLEF2019 evaluation campaign. We report system performance for the campaign's two subtasks, humour detection and funniness score prediction, and discuss some issues arising from the conversion between the numeric scores used in the HAHA@IberLEF2019 data and the pairwise judgment annotations required for our method.

Keywords: Computational humour, humour, Gaussian process preference learning, GPPL, best–worst scaling

Resumen: Actualmente la mayoría de los sistemas de procesamiento de humor hacen, en el mejor de los casos, distinciones discretas y granualeres entre lo cómico y lo convencional. Sin embargo, dichos conceptos se conciben mejor en un espectro más amplio. Este artículo presenta un método probabilístico, un modo de preferencias de aprendizaje basadas en un proceso gaussiano (GPPL), que aprende a clasificar y calificar el humor de textos cortos explotando juicios de preferencia humana y anotaciones lingüísticas generadas en forma automática. Nuestro sistema es similar a uno que previamente había demostrado un buen desempeño en frases en inglés anotadas con anotaciones humorísticas por pares y lo aplicamos a la colección de datos en español de la campaña de evaluación HAHA@IberLEF2019. En este trabajo reportamos el desempeño del sistema para dos subtareas de la campaña: detección de humor y predicción de puntaje de diversión. También presentamos algunos problemas que surgen de la conversión entre los puntajes numéricos utilizados en los datos HAHA@IberLEF2019 y las anotaciones de juicio de pares de documentos requeridas para nuestro método.

Palabras clave: Humor computacional, humor, preferencias de aprendizaje, proceso gaussiano, GPPL, mejor-peor escala

ISSN 1135-5948. DOI 10.26342/2020-64-4

1 Introduction

Humour is an essential part of everyday communication, particularly in social media (Holton and Lewis, 2011; Shifman, 2013), yet it remains a challenge for computational methods. Unlike conventional language, humour requires complex linguistic and background knowledge to understand, which are difficult to integrate with NLP methods (Hempelmann, 2008).

An important step in the automatic processing of humour is to recognize its presence in a piece of text. However, its intensity may be present or perceived to varying degrees to its human audience (Bell, 2017). This level of appreciation (i.e., *humorousness* or equivalently *funniness*) can vary according to the text's content and structural features, such as nonsense or disparagement (Carretero-Dios, Pérez, and Buela-Casal, 2010) or, in the case of puns, contextual coherence (Lippman and Dunn, 2000) and the cognitive effort required to recover the target word (Hempelmann, 2003, pp. 123–124).

While previous work has considered mainly binary classification approaches to humorousness, the HAHA@IberLEF2019 shared task (Chiruzzo et al., 2019) also focuses on its gradation. This latter task is important for downstream applications such as conversational agents or machine translation, which must choose the correct tone in response to humour, or find appropriate jokes and wordplay in a target language. The degree of creativeness may also inform an application whether the semantics of a joke can be inferred from similar examples.

This paper describes a system designed to carry out both subtasks of the HAHA@IberLEF2019 evaluation campaign: binary classification of tweets as humorous or not humorous, and the quantification of humour in those tweets. Our system employs a Bayesian approach—namely, a variant of Gaussian process preference learning (GPPL) that infers humorousness scores or rankings on the basis of manually annotated pairwise preference judgments and automatically annotated linguistic features. In the following sections, we describe and discuss the background and methodology of our system, our means of adapting the HAHA@IberLEF2019 data to work with our system, and the results of our system evaluation on this data.

2 Background

Pairwise comparisons can be used to infer rankings or ratings by assuming a *random utility model* (Thurstone, 1927), meaning that the annotator chooses an instance (from a pair or list of instances) with probability p, where p is a function of the *utility* of the instance. Therefore, when instances in a pair have similar utilities, the annotator selects one with a probability close to 0.5, while for instances with very different utilities, the instance with higher utility will be chosen consistently. The random utility model forms the core of two popular preference learning models, the Bradley–Terry model (Bradley and Terry, 1952; Luce, 1959; Plackett, 1975), and the Thurstone–Mosteller model (Thurstone, 1927; Mosteller, 1951). Given this model and a set of pairwise annotations, probabilistic inference can be used to retrieve the latent utilities of the instances.

Besides pairwise comparisons, a random utility model is also employed by MaxDiff (Marley and Louviere, 2005), a model for best–worst scaling (BWS), in which the annotator chooses the best and worst instances from a set. While the term "best–worst scaling" originally applied to the data collection technique (Finn and Louviere, 1992), it now also refers to models such as MaxDiff that describe how annotators make discrete choices. Empirical work on BWS has shown that MaxDiff scores (instance utilities) can be inferred using either maximum likelihood or a simple counting procedure that produces linearly scaled approximations of the maximum likelihood scores (Flynn and Marley, 2014). The counting procedure defines the score for an instance as the fraction of times the instance was chosen as best, minus the fraction of times the instance was chosen as worst, out of all comparisons including that instance (Kiritchenko and Mohammad, 2016). From this point on, we refer to the counting procedure as BWS, and apply it to the tasks of inferring scores from pairwise annotations for funniness.

Gaussian process preference learning (GPPL) (Chu and Ghahramani, 2005), a Thurstone–Mosteller–based model that accounts for the features of the instances when inferring their scores, can make predictions for unlabelled instances and copes better with sparse pairwise labels. GPPL uses Bayesian inference, which has been shown to cope better with sparse and noisy data (Xiong, Barash, and Frey, 2011; Titov and Klementiev, 2012; Beck, Cohn, and Specia, 2014; Lampos et al., 2014), including disagreements between multiple annotators (Cohn and Specia, 2013; Simpson et al., 2015; Felt, Ringger, and Seppi, 2016; Kido and Okamoto, 2017). Through the random utility model, GPPL handles disagreements between annotators as noise, since no instance in a pair has a probability of one of being selected.

Given a set of pairwise labels, and the features of labelled instances, GPPL can estimate the posterior distribution over the utilities of any instances given their features. Relationships between instances are modelled by a Gaussian process, which computes the covariance between instance utilities as a function of their features (Rasmussen and Williams, 2006). Since typical methods for posterior inference (Nickisch and Rasmussen, 2008) are not scalable (the computational complexity is $\mathcal{O}(n^3)$, where n is the number of instances), we use a scalable method for GPPL that permits arbitrarily large numbers of instances and pairs (Simpson and Gurevych, 2018). This method uses stochastic variational inference (Hoffman et al., 2013), which limits computational complexity by substituting the instances for a fixed number of *inducing points* during inference.

The GPPL method has already been applied with good results to ranking arguments by convincingness (which, like funniness, is an abstract linguistic property that is hard to quantify directly) and to ranking English-language one-liners by humorousness (Simpson and Gurevych, 2018; Simpson et al., 2019). In these two tasks, GPPL was found to outperform SVM and BiLSTM regression models that were trained directly on gold-standard scores, and to outperform BWS when given sparse training data, respectively. We therefore elect to use GPPL on the Spanish-language Twitter data of the HAHA@IberLEF2019 shared task.

In the interests of replicability, we freely release the code for running our GPPL system, including the code for the data conversion and subsampling process detailed in §3.2.[1]

3 Experiments

3.1 Tasks

The HAHA@IberLEF2019 evaluation campaign consists of two tasks. Task 1 is humour detection, where the goal is to predict whether or not a given tweet is humorous, as determined by a gold standard of binary, human-sourced annotations. Systems are scored on the basis of accuracy, precision, recall, and F-measure. Task 2 is humorousness prediction, where the aim is to assign each funny tweet a score approximating the average funniness rating, on a five-point scale, assigned by a set of human annotators. Here system performance is measured by root-mean-squared error (RMSE). For both tasks, the campaign organizers provide a collection of 24 000 manually annotated training examples. The test data consists of a further 6000 tweets whose gold-standard annotations were withheld from the participants.

3.2 Data Preparation

For each of the 24 000 tweets in the HAHA@IberLEF2019 training data, the task organizers asked human annotators to indicate whether the tweet was humorous, and if so, how funny they found it on a scale from 1 (“not funny”) to 5 (“excellent”). This is essentially the same annotation scheme used for the first version of the corpus (Castro et al., 2018) which was used in the previous iteration of HAHA (Castro, Chiruzzo, and Rosá, 2018). As originally distributed, the training data gives the text of each tweet along with the number of annotators who rated it as “not humour”, “1”, “2”, “3”, “4”, and “5”. For the purposes of Task 1, tweets in the positive class received at least three numerical annotations and at least five annotations in total; tweets in the negative class received at least three “not humour” annotations, though possibly fewer than five annotations in total. Only those tweets in the positive class are used in Task 2.

This ordinal data cannot be used as-is with our GPPL system, which requires as input a set of preference judgments between pairs of instances. To work around this, we converted the data into a set of ordered pairs of tweets such that the first tweet has a lower average funniness score than the second. (We consider instances in the negative class to have an average funniness score of 0.) While an exhaustive set of pairings would contain 575 976 000 pairs

[1] `https://github.com/UKPLab/haha2019-GPPL`

(minus the pairs in which both tweets have the same score), we produced only 10 730 229 pairs, which was the minimal set necessary to accurately order the tweets. For example, if the original data set contained three tweets A, B, and C with average funniness scores 5.0, 3.0, and 1.0, respectively, then our data would contain the pairs (C, B) and (B, A) but not (C, A). To save memory and computation time in the training phase, we produced a random subsample such that the number of pairs where a given tweet appeared as the funnier one was capped at 500. This resulted in a total of 485 712 pairs. In a second configuration, we subsampled up to 2500 pairs per tweet. We used a random 60% of this set to meet memory limitations, resulting in 686 098 pairs.

With regards to the tweets' textual data, we do only basic tokenization as preprocessing. For lookup purposes (synset lookup; see §3.3), we also lemmatize the tweets.

3.3 Experimental Setup

As we adapt an existing system that works on English data (Simpson et al., 2019), we generally reuse the features employed there, but use Spanish resources instead. Each tweet is represented by the vector resulting from a concatenation of the following:

- The average of the word embedding vectors of the tweet's tokens, for which we use 200-dimensional pretrained Spanish Twitter embeddings (Deriu et al., 2017).
- The average frequency of the tweet's tokens, as determined by a Wikipedia dump.[2]
- The average word polysemy—i.e., the number of synsets per lemma of the tweet's tokens, as given by the Multilingual Central Repository (MCR 3.0, release 2016) (Gonzalez-Agirre, Laparra, and Rigau, 2012).

Using the test data from the HAHA@IberLEF2018 task (Castro, Chiruzzo, and Rosá, 2018) as a development set, we further identified the following features from the UO_UPV system (Ortega-Bueno et al., 2018) as helpful:

- The heuristically estimated turn count (i.e., the number of tokens beginning with `-` or `--`) and binary dialogue heuristic (i.e., whether the turn count is greater than 2).
- The number of hashtags (i.e., tokens beginning with `#`).
- The number of URLs (i.e., tokens beginning with `www` or `http`).
- The number of emoticons.[3]
- The character and token count, as well as mean token length.
- The counts of exclamation marks and other punctuation (`.,;?`).

We adapt the existing GPPL implementation[4] using the authors' recommended hyperparameter defaults (Simpson and Gurevych, 2018): batch size $|P_i| = 200$, scale hyperparameters $\alpha_0 = 2$ and $\beta_0 = 200$, and the number of inducing points (i.e., the smaller number of data points that act as substitutes for the tweets in the dataset) $M = 500$. The maximum number of iterations was set to 2000. Using these feature vectors, hyperparameter settings, and data pairs, we require a training time of roughly two hours running on a 24-core cluster with 2 GHz CPU cores.

After training the model, an additional step is necessary to transform the GPPL output values to the original funniness range (0, 1–5). For this purpose, we train a Gaussian process regressor which we supply with the output values of the GPPL system as features and the corresponding HAHA@IberLEF2018 test data values as targets. However, this model can still yield results outside the desired range when applied to the GPPL output of the HAHA@IberLEF2019 test data. Thus, we afterwards map too-large and too-small values onto the range boundaries. We further set an empirically determined threshold for binary funniness estimation.

3.4 Results and Discussion

Table 1 reports results for the binary classification setup (Task 1) and the regression task (Task 2). Included in each table are the scores

[2] `https://dumps.wikimedia.org/eswiki/20190420/eswiki-20190420-pages-articles.xml.bz2`; last accessed on 2019-06-15.

[3] `https://en.wikipedia.org/wiki/List_of_emoticons\#Western`, Western list; last accessed on 2019-06-15.

[4] `https://github.com/UKPLab/tacl2018-preference-convincing`

	Task 1				Task 2
System	F$_1$	Precision	Recall	Accuracy	RMSE
Ismailov (2019)	0.821	0.791	0.852	0.855	0.736
our system	0.660	0.588	0.753	0.698	1.810
baseline	0.440	0.394	0.497	0.505	2.455

Table 1: Results for Task 1 (humour detection) and Task 2 (funniness score prediction)

of our own system, as well as those of the top-performing system (Ismailov, 2019) and a naïve baseline. For Task 1, the naïve baseline makes a random classification for each tweet (with uniform distribution over the two classes); for Task 2, it assigns a funniness score of 3.0 to each tweet.

In the binary classification setup, our system achieved an F-measure of 0.660 on the test data, representing a precision of 0.588 and a recall of 0.753. In the regression task, we achieved RMSE of 1.810. The results are based on the second data subsample (up to 2500 pairs), with the results for the first (up to 500 pairs) being slightly lower. Our results for both tasks, while handily beating those of the naïve baseline, are significantly worse than those reported by some other systems in the evaluation campaign, including of course the winner. This is somewhat surprising given GPPL's very good performance in previous English-language experiments (Simpson et al., 2019).

Unfortunately, our lack of fluency in Spanish and lack of access to the gold-standard scores for the test set tweets precludes us from performing a detailed qualitative error analysis. However, we speculate that our system's less than stellar performance can partly be attributed to the information loss in converting between the numeric scores used in the HAHA@IberLEF2019 tasks and the preference judgments used by our GPPL system. In support of this explanation, we note that the output of our GPPL system is rather uniform; the scores occur in a narrow range with very few outliers. (Figure 1 shows this outcome for the HAHA@IberLEF2018 test data.) Possibly this effect would have been less pronounced had we used a much larger subsample, or even the entirety, of the possible training pairs, though as discussed in §3.2, technical and temporal limitations prevented us from doing so. We also speculate that the Gaussian process regressor we used may not have been the best way of mapping our GPPL scores back onto the task's funniness scale (albeit still better than a linear mapping).

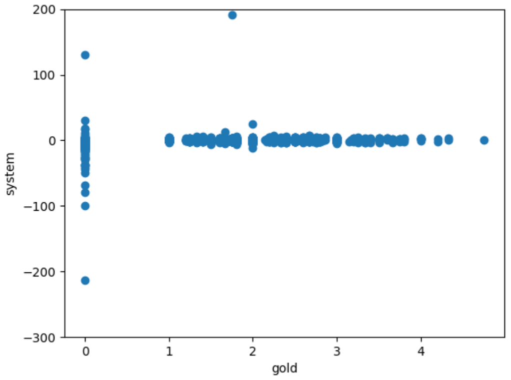


Figure 1: Gold values of the HAHA@IberLEF2018 test data (x axis) vs. the scores assigned by our GPPL system (y axis), before mapping to the expected funniness range using a Gaussian process regressor. The lowest GPPL value (-1400) was removed from the plot to obtain a better visualization.

Apart from the difficulties posed by the differences in the annotation and scoring, our system may have been affected by the mismatch between the language resources for most of its features and the language of the test data. That is, while we relied on language resources like Wikipedia and MCR that reflect standardized registers and prestige dialects, the HAHA@IberLEF2019 data is drawn from unedited social media, whose language is less formal, treats a different range of topics, and may reflect a wider range of dialects and writing styles. Twitter data in particular is known to present problems for vanilla NLP systems, at least without extensive cleaning and normalization (W-NUT, 2015). This is reflected in our choice of word embeddings: while we achieved a Spearman rank correlation of $\rho = 0.52$ with the HAHA@IberLEF2018 test data using embeddings based on Twitter data (Deriu et al., 2017), the same system using more "standard" Wikipedia-/news-/Web-based embeddings[5] resulted in a correlation near zero.

4 Conclusion

This paper has presented a system for predicting both binary and graded humorousness. It employs Gaussian process preference learning, a Bayesian system that learns to rank and rate instances by exploiting pairwise preference judgments. By providing additional feature data (in our case, shallow linguistic features), the method can learn to predict scores for previously unseen items.

Though our system is based on one that had previously achieved good results with rudimentary, task-agnostic linguistic features on two English-language tasks (including one involving the gradation of humorousness), its performance on the Spanish-language Twitter data of HAHA@IberLEF2019 was less impressive. We tentatively attribute this to the information loss involved in the (admittedly artificial) conversion between the numeric annotations used in the task and the preference judgments required as input to our method, and to the fact that we do not normalize the Twitter data to match our linguistic resources.

A possible avenue of future work, therefore, might be to mitigate the data conversion problem. However, as it has been rather convincingly argued, both generally (Thurstone, 1927) and in the specific case of humour assessment (Shahaf, Horvitz, and Mankoff, 2015), that aggregate ordinal rating data should not be treated as interval data, the proper solution here would be to recast the entire task from one of binary classification or regression to one of comparison or ranking. Perhaps the best way of doing this would be to source new gold-standard preference judgments on the data, though this would be an expensive and time-consuming endeavour.[6]

Regardless of the task setup, there are a few further ways our system might be improved. First, we might try normalizing the language of the tweets, and secondly, we might try using additional, humour-specific features, including some of those used in past work as well as those inspired by the prevailing linguistic theories of humour (Attardo, 1994). The benefits of including word frequency also point to possible further improvements using n-grams, TF–IDF, or other task-agnostic linguistic features.

Acknowledgments

The authors thank Gisela Vallejo for her help with Spanish-language translations.

This work has been supported by the German Federal Ministry of Education and Research (BMBF) under the promotional reference 01UG1816B (CEDIFOR), by the German Research Foundation (DFG) as part of the QA-EduInf project (grants GU 798/18-1 and RI 803/12-1), by the DFG-funded research training group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES; GRK 1994/1), and by the Austrian Science Fund (FWF) under project M 2625-N31. The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry for Science, Research and Economy.

References

Attardo, S. 1994. *Linguistic Theories of Humor.* Mouton de Gruyter, Berlin.

Beck, D., T. Cohn, and L. Specia. 2014. Joint emotion analysis via multi-task Gaussian processes. In *Proc. 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1798–1803.

Bell, N. D. 2017. Failed humor. In S. Attardo, editor, *The Routledge Handbook of Language and Humor.* Routledge, New York, pages 356–370.

Bradley, R. A. and M. E. Terry. 1952. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 39(3/4):324–345, December.

Carretero-Dios, H., C. Pérez, and G. Buela-Casal. 2010. Assessing the appreciation of the content and structure of humor: Construction of a new scale. *Humor*, 23(3):307–325, August.

Castro, S., L. Chiruzzo, and A. Rosá. 2018. Overview of the HAHA task: Humor analysis based on human annotation at IberEval 2018. In *Proc. Third Workshop on Evaluation of Human Language Technologies for Iberian Languages*, volume 2150 of *CEUR Workshop Proceedings*, pages 187–194.

[5] `https://zenodo.org/record/1410403`

[6] We note with appreciation that the upcoming SemEval-2020 task on humour assessment (Hossain et al., 2020) does include a subtask for predicting preference judgments, though it seems the underlying gold-standard data still uses aggregate ordinal data.

Castro, S., L. Chiruzzo, A. Rosá, D. Garat, and G. Moncecchi. 2018. A crowd-annotated Spanish corpus for humor analysis. In *Proc. Sixth International Workshop on Natural Language Processing for Social Media*, pages 7–11.

Chiruzzo, L., S. Castro, M. Etcheverry, D. Garat, J. J. Prada, and A. Rosá. 2019. Overview of HAHA at IberLEF 2019: Humor analysis based on human annotation. In *Proc. Iberian Languages Evaluation Forum*, volume 2421 of *CEUR Workshop Proceedings*, pages 132–144.

Chu, W. and Z. Ghahramani. 2005. Preference learning with Gaussian processes. In *Proc. 22nd International Conference on Machine Learning*, pages 137–144.

Cohn, T. and L. Specia. 2013. Modelling annotator bias with multi-task Gaussian processes: An application to machine translation quality estimation. In *Proc. 51st Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 32–42.

Deriu, J., A. Lucchi, V. De Luca, A. Severyn, S. Müller, M. Cieliebak, T. Hoffmann, and M. Jaggi. 2017. Leveraging large amounts of weakly supervised data for multi-language sentiment classification. In *Proc. 26th International World Wide Web Conference*, pages 1045–1052.

Felt, P., E. K. Ringger, and K. D. Seppi. 2016. Semantic annotation aggregation with conditional crowdsourcing models and word embeddings. In *Proc. 26th International Conference on Computational Linguistics*, pages 1787–1796.

Finn, A. and J. J. Louviere. 1992. Determining the appropriate response to evidence of public concern: The case of food safety. *J. Public Policy & Market.*, 11(2):12–25, September.

Flynn, T. N. and A. A. J. Marley. 2014. Best–worst scaling: Theory and methods. In S. Hess and A. Daly, editors, *Handbook of Choice Modelling*. Edward Elgar Publishing, Cheltenham, UK, pages 178–201.

Gonzalez-Agirre, A., E. Laparra, and G. Rigau. 2012. Multilingual Central Repository version 3.0. In *Proc. 8th International Conference on Language Resources and Evaluation*, pages 2525–2529.

Hempelmann, C. F. 2003. *Paronomasic Puns: Target Recoverability Towards Automatic Generation*. Ph.D. thesis, Purdue University, West Lafayette, IN, USA.

Hempelmann, C. F. 2008. Computational humor: Beyond the pun? In V. Raskin, editor, *The Primer of Humor Research*. Mouton de Gruyter, Berlin, pages 333–360.

Hoffman, M. D., D. M. Blei, C. Wang, and J. W. Paisley. 2013. Stochastic variational inference. *J. Mach. Learn. Res.*, 14:1303–1347, May.

Holton, A. E. and S. C. Lewis. 2011. Journalists, social media, and the use of humor on Twitter. *Electron. J. Commun.*, 21(1&2).

Hossain, N., J. Krumm, M. Gamon, and H. Kautz. 2020. SemEval-2020 Task 7: Assessing the funniness of edited news headlines. In *Proc. 14th International Workshop on Semantic Evaluation*. To appear.

Ismailov, A. 2019. Humor analysis based on human annotation challenge at IberLEF 2019: First-place solution. In *Proc. Iberian Languages Evaluation Forum*, volume 2421 of *CEUR Workshop Proceedings*, pages 160–164.

Kido, H. and K. Okamoto. 2017. A Bayesian approach to argument-based reasoning for attack estimation. In *Proc. 26th International Joint Conference on Artificial Intelligence*, pages 249–255.

Kiritchenko, S. and S. M. Mohammad. 2016. Capturing reliable fine-grained sentiment associations by crowdsourcing and best–worst scaling. In *Proc. 2016 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 811–817.

Lampos, V., N. Aletras, D. Preoţiuc-Pietro, and T. Cohn. 2014. Predicting and characterising user impact on Twitter. In *Proc. 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 405–413.

Lippman, L. G. and M. L. Dunn. 2000. Contextual connections within puns: Effects on perceived humor and memory. *J. Gen. Psychol.*, 127(2):185–197, April.

Luce, R. D. 1959. On the possible psychophysical laws. *Psychol. Rev.*, 66(2):81–95.

Marley, A. A. J. and J. J. Louviere. 2005. Some probabilistic models of best, worst, and best–worst choices. *J. Math. Psychol.*, 49(6):464–480.

Mosteller, F. 1951. Remarks on the method of paired comparisons: I. The least squares solution assuming equal standard deviations and equal correlations. *Psychometrika*, 16(1):3–9, March.

Nickisch, H. and C. E. Rasmussen. 2008. Approximations for binary Gaussian process classification. *J. of Mach. Learn. Res.*, 9:2035–2078, October.

Ortega-Bueno, R., C. E. Muñiz Cuza, J. E. Medina Pagola, and P. Rosso. 2018. UO_UPV: Deep linguistic humor detection in Spanish social media. In *Proc. Third Workshop on Evaluation of Human Language Technologies for Iberian Languages*, volume 2150 of *CEUR Workshop Proceedings*, pages 203–213.

Plackett, R. L. 1975. The analysis of permutations. *J. Royal Stat. Soc. Ser. C (Appl. Stat.)*, 24(2):193–202.

Rasmussen, C. E. and C. K. I. Williams. 2006. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, USA.

Shahaf, D., E. Horvitz, and R. Mankoff. 2015. Inside jokes: Identifying humorous cartoon captions. In *Proc. 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1065–1074.

Shifman, L. 2013. *Memes in Digital Culture*. MIT Press, Cambridge, MA, USA.

Simpson, E., E.-L. Do Dinh, T. Miller, and I. Gurevych. 2019. Predicting humorousness and metaphor novelty with Gaussian process preference learning. In *Proc. 57th Annual Meeting of the Association for Computational Linguistics*, pages 5716–5728.

Simpson, E. and I. Gurevych. 2018. Finding convincing arguments using scalable Bayesian preference learning. *Trans. Assoc. Comput. Ling.*, 6:357–371.

Simpson, E. D., M. Venanzi, S. Reece, P. Kohli, J. Guiver, S. J. Roberts, and N. R. Jennings. 2015. Language understanding in the wild: Combining crowdsourcing and machine learning. In *Proc. 24th International Conference on World Wide Web*, pages 992–1002.

Thurstone, L. L. 1927. A law of comparative judgment. *Psychol. Rev.*, 34(4):273–286.

Titov, I. and A. Klementiev. 2012. A Bayesian approach to unsupervised semantic role induction. In *Proc. 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 12–22.

2015. *Proc. ACL 2015 Workshop on Noisy User-generated Text*. Association for Computational Linguistics.

Xiong, H. Y., Y. Barash, and B. J. Frey. 2011. Bayesian prediction of tissue-regulated splicing using RNA sequence and cellular context. *Bioinform.*, 27(18):2554–2562, September.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 45-52 recibido 18-11-2019 revisado 08-01-2020 aceptado 11-01-2020

Building wordnets with multi-word expressions from parallel corpora

Expansión de wordnets mediante unidades pluriverbales extraídas de corpus paralelos

Alberto Simões[1], **Xavier Gómez Guinovart**[2]
[1]2Ai – School of Technology, IPCA, Barcelos, Portugal
[2]Universidade de Vigo, SLI-TALG
asimoes@ipca.pt, xgg@uvigo.gal

Abstract: In this paper we present a method for enlarging wordnets focusing on multi-word terms and utilising data from parallel corpora. Our approach is validated using the Galician and Portuguese wordnets. The multi-word candidates obtained in this experiment were manually validated, obtaining a 73.2% accuracy for the Galician language and a 75.5% for the Portuguese language.
Keywords: wordnet, parallel corpora, lexical resources, multi-word expressions

Resumen: Presentamos un método para la ampliación de wordnets en el ámbito de las unidades pluriverbales, usando datos de corpus paralelos y aplicando el método a la expansión de los wordnets del gallego y del portugués. Las unidades pluriverbales que se obtienen en este experimento se validaron manualmente, obteniendo una precisión del 73.2% para el gallego y del 75.5% para el portugués.
Palabras clave: wordnet, corpus paralelos, recursos léxicos, unidades pluriverbales

1 Introduction

The Princeton WordNet (Miller et al., 1990)[1] (PWN) is, undoubtedly, a milestone in Natural Language Processing. This can be proven by the amount of wordnet-like projects available for most of the world languages. In this article we will discuss a methodology to enrich two different wordnets: Galnet (Gómez Guinovart and Solla Portela, 2018)[2] for Galician, and PULO (Simões and Gómez Guinovart, 2013)[3] for Portuguese.

Given the amount of manual work required to produce a quality resource, there have been different approaches to create new wordnets. Our proposal focuses only on the multi-word noun entries in wordnet, and on the usage of parallel corpora and translation patterns to obtain candidates for the target language.

This paper is structured as follows: Section 2 summarises the previous related work. Section 3 presents the resources used (wordnets, parallel corpora) and the derived resources (multi-word term list, annotated corpora, and probabilistic translation dictionaries). There follows Section 4 where the implemented algorithm is explained, and Section 5 where an evaluation of the obtained candidates is performed. Finally we draw some conclusions and present some directions for future work.

2 Related Work

Galnet and PULO wordnets have been created from the English WordNet 3.0, following the expand model (Vossen, 1998), where the variants associated with the PWN synsets are obtained through different strategies. The same approach has been taken in the MCR framework (González-Agirre and Rigau, 2013) for the creation of the wordnets of Spanish, Catalan and Basque.

One of the main methodologies used to extend a wordnet coverage from the variants associated with the PWN synsets is the acquisition of their translations from parallel corpora. Thus, in (Gómez Guinovart and Oliver, 2014) the authors apply that methodology to expand the Galnet first distribution from two different available parallel

[1] https://wordnet.princeton.edu
[2] http://sli.uvigo.gal/galnet/
[3] http://wordnet.pt

ISSN 1135-5948. DOI 10.26342/2020-64-5

textual resources: the automatically translated English–Galician SemCor Corpus;[4] and the English–Galician and Spanish–Galician sections of the CLUVI Corpus.[5] In either case, only the English or Spanish part of the parallel corpora has been sense-tagged for the experiment. In (Oliver, 2014) the same methodology is applied to the automatic translation of the English SemCor to six languages (Catalan, Spanish, French, German, Italian and Portuguese).

In (Simões and Gómez Guinovart, 2018), the authors used the Galician, Portuguese, Spanish, Catalan and English versions of the Bible from the CLUVI Corpus. They were annotated with part-of-speech and WordNet sense. The resulting synsets were aligned, and new variants for Galnet were extracted. After manual evaluation the approach presented a 96.8% accuracy. Unlike the research we present in this paper, all these previous experiments have been focused on monolexical extraction.

In (Gómez Guinovart and Simões, 2009) the authors presented a parallel corpora-based bilingual terminology extraction method based on the occurrence of bilingual morphosyntactic patterns in parallel text, with the support of probabilistic translation dictionaries for inter-language alignment. We applied this method using corpora for English–Galician and English–Portuguese, obtaining an accuracy rate between 87.4% and 96% depending on the characteristics of the corpus.

(Vintar and Fišer, 2008) present an approach to extend the automatically created Slovene wordnet with nominal multi-words from the English-Slovene part of the JRC-Acquis corpus of legislative text of the European Union by translating multi-words from Princeton WordNet with a technique that is based on word alignment and lexico-syntactic patterns. For each source multi-word, they extracted all sentence pairs from the parallel corpus that contain the source term. Also, for each single word from the source multi-word they extract all possible translation equivalents from the bilingual lexicon. Then, they use lexico-grammatical patterns to identify potential multi-word terms in the target language and check the word alignments for the selection of the best equivalent, which is the candidate with the most matches for each constituent word in the bilingual lexicon. The authors manually evaluated the set of candidate words obtained by this technique, filtered by a threshold of 0.05 as the lowest possible similarity score, obtaining an accuracy of 85% and a total of 1,059 new variants for the Slovene wordnet.

3 Resources

As pointed out before, the methodology we propose requires a source wordnet and a parallel corpus mapping the source wordnet language to the target language.

3.1 Wordnets

Both Galnet and PULO are part of the Multilingual Central Repository (MCR),[6] that currently integrates wordnets from six different languages (English, Spanish, Catalan, Galician, Basque and Portuguese) with WordNet 3.0 as Interlingual Index (ILI) (González-Agirre and Rigau, 2013). Table 1 provides the number of synsets and variants for the different languages gathered in this repository, and their percentage of development with respect to the English WordNet.

From the English WordNet a list of multi-word terms were extracted (Lloberes et al., 2013). There are 68,751 multi-word terms in the PWN. We decided to focus our experiment on the noun terms (63,073, above 90% of the total amount of multi-word terms). This list was then processed by FreeLing 4.1 (Padró and Stanilovsky, 2012)[7] in order to obtain each term's morphological structure, and understand which of these structures are more common and more likely to return interesting results.

3.2 Parallel corpora

Parallel corpora were obtained from the OPUS project.[8] For the English–Galician pair, we used the Gnome, KDE 4, Ubuntu, Tatoeba and OpenSubtitles2018 corpora, amounting to a total of 350,124 translation units. Given the limited existence of English–Galician corpora, there was no other reasonable alternative. For the English–Portuguese pair, only the OpenSubtitles2018 corpus was used, accounting for 26,805,614 translation units.

[4] `http://gabormelli.com/RKB/SemCor_Corpus`

[5] `http://sli.uvigo.gal/CLUVI/`

[6] `http://adimen.si.ehu.es/web/MCR/`

[7] `http://nlp.lsi.upc.edu/freeling/`

[8] `http://opus.nlpl.eu`

	English (PWN 3.0)		Galician (Galnet 3.0.28)	
	variants	synsets	variants	synsets
Total	206,941	117,659	70,056	43,057
%	100%	100%	33.8%	36.6%
	Spanish (MCR 2016)		Portuguese (MCR 2016)	
Total	146,501	78,995	32,604	17,942
%	70.8%	67.1%	15.8%	15.2%
	Catalan (MCR 2016)		Basque (MCR 2016)	
Total	100,793	60,956	50,037	30,263
%	48.7%	51.8%	24.2%	25.7%

Table 1: Current coverage of languages in MCR

These corpora were processed in two different ways:

- Lemmatisation and part-of-speech annotation using FreeLing. This annotation was performed without any kind of named entity or locution detection in the target language, so that the resulting corpus does not include any multi-word terms annotated. For the source language (English), the corpus was only tagged with the FreeLing multi-word recognition module using the list of multi-word expressions referred to in the previous section.
- Both the original corpus as the lemmatised corpus (result of the above annotation process) were subject of word-alignment using NATools (Simões and Almeida, 2003). Thus, for each language pair we obtained a probabilistic translation dictionary (PTD) for forms and lemmas. Each entry of a PTD maps a word in the source language to a set of probable translations as well as their translation probability.

4 Methodology

Our approach requires not just the resources presented in the previous section but also a set of morphosyntactic patterns. These patterns will later be used in the extraction algorithm to obtain variant candidates.

4.1 Morphosyntactic patterns

As pointed out before, this approach uses translation patterns: rules that make explicit the number of words (tokens) in each language, how the translation of each word switches its place during the translation process, and whether there is any addition or removal of words.

The rules were created starting from the list of multi-word terms present in the English WordNet, together with their morphological structure. Only the top 10 occurring structures were chosen for this experiment. A set of multi-word terms following each one of the morphological structures was compiled in order to manually study how their translation was performed. This study resulted in the patterns for English–Galician and for English–Portuguese presented in Figure 1, which cover about 90% of the multi-word nouns in WordNet. Note that FreeLing uses two different tag sets for English and for Portuguese/Galician, and that is why the prefixes presented on the left-hand side and the right-hand-side of the rule are different. Note also that both Galician and Portuguese share the same multi-word translation patterns for English–Galician and for English–Portuguese as a result of their similar morphological structures.

In the rules file, everything starting with a sharp (#) character is considered a comment. Then, each line is comprised of a left-hand side pattern, matching the English variant, and a right-hand side pattern that will try to match the corresponding Galician or Portuguese variant. Each item (token) in the pattern is separated from each other by a space. Each token can have a name (upper-case before the parenthesis) or a lemma (lower-case before the parenthesis), but never both, given that identifiers are used to match translations and, if a specific lemma is supplied, there is no translation check. Inside the parenthesis is the morphological category of the token being matched (using the beginning segment of any FreeLing tag).

```
#1. air compressor = compresor de aire / compressor de ar
A(NN) B(NN) =  B(NC) de(SP) o(DA)? A(NC)
#2. absolute zero = cero absoluto / zero absoluto
A(JJ) B(NN) =  B(NC) A(AQ)
#3. lubricating system = sistema de lubricación / sistema de lubrificação
A(VBG) B(NN) =  B(NC) de(SP) o(DA)? A(NC)
#4. analysis of variance = análise de varianza / análise de variância
A(NN) B(IN) C(NN) =  A(NC) B(SP) o(DA)? C(NC)
#5. closed circuit = circuíto pechado / circuito fechado
A(VBN) B(NN) = B(NC) A(VMP)
#6. communications satellite = satélite de comunicacións / satélite de comunicações
A(NNS) B(NN) =  B(NC) de(SP) o(DA)? A(NC)
#7. local area network = rede de área local / rede de área local
A(JJ) B(NN) C(NN) = C(NC) de(SP) o(DA)? B(NC) A(AQ)
#8. graphical user interface = interface gráfica de usuario / interface gráfica do utilizador
A(JJ) B(NN) C(NN) = C(NC) A(AQ) de(SP) o(DA)? B(NC)
#9. African green monkey = mono verde africano / macaco verde africano
A(JJ) B(JJ) C(NN) = C(NC) B(AQ) A(AQ)
#10. table of contents = táboa de contidos / tabela de conteúdo
A(NN) B(IN) C(NNS) = A(NC) B(SP) o(DA)? C(NC)
```

Figure 1: English–Galician/Portuguese rules and examples

As can be seen in the following specific pattern:

A(NN) B(NN) = B(NC) de(SP) o(DA)? A(NC)

the left-hand side is matching two nouns (tags starting with `NN`). The first one would be identified by A, and the second one by B. Therefore, considering the multi-word "*air compressor*" the following would be extracted:

$$\{A \mapsto \textit{air}, B \mapsto \textit{compressor}\}$$

The right-hand side of the rule specifies that the algorithm should look for a sequence of a noun (`NC`), a preposition (`SP`), an optional article (`DA`) and another noun. Note that the question mark following a token specifies that it is optional. Together with this sequence, the right-hand side also specifies that the first token found should be the translation of B, following by a token with lemma '*de*', another token with lemma '*o*', and finally a token that should be the translation of A.

4.2 Matching algorithm

The first part of the process is to create a reverse index. This index maps each nominal phrase from the list of multi-word terms to the translation units of the parallel corpora where they occur. By "translation unit" we refer to a pair of source and target sentences in the corpus, whether English–Galician or English–Portuguese.

Then, for each pair $(synset, variant)$ the following process is executed:

1. Ignore the pair if the source variant structure does not match any of the defined translation patterns.

2. For each translation pattern matching the source variant, search in the sequence of parts-of-speech for the target language if there is any occurrence for the right-hand side of the translation pattern.

3. If there is one or more occurrences of the target pattern, each one is evaluated, checking the probable translation probability with the source variant.

4. To evaluate the translation probability the rule placeholder identifiers come into play. The translation probability is computed for the words associated with the identifiers (for the source and target language) both for forms and lemmas. The same is done in the reverse order, as the probabilistic translation dictionaries are not symmetrical. For example, the pair (*air compressor*, *compresores de aire*) has $\{\text{A} \mapsto \textit{air}, \text{B} \mapsto \textit{compressor}\}$ for the source language, and $\{\text{A} \mapsto \textit{aire}, \text{B} \mapsto \textit{compressores}\}$ for the target language.

   Its translation probability in the source–

target direction is computed by:[9]

$$\begin{aligned}\mathcal{P} &= \frac{1}{4}\mathcal{P}_l\left(\mathcal{T}(air) = aire\right)\\ &+ \frac{1}{4}\mathcal{P}_f\left(\mathcal{T}(air) = aire\right)\\ &+ \frac{1}{4}\mathcal{P}_l\left(\mathcal{T}(compressor) = compresores\right)\\ &+ \frac{1}{4}\mathcal{P}_f\left(\mathcal{T}(compressor) = compresor\right)\end{aligned}$$

The same is done in the reverse direction, using the GL–EN and PT–EN dictionaries. The two probabilities obtained are then averaged.

5. Given all possible PoS alignments with the target sentence, only the one with greatest probability is kept.
6. Finally for all occurrences of the original variant, the target sequence whose alignment occurred more times is chosen.
7. In the final list of candidates, only the ones with probability greater than 0.1 were considered. This threshold was defined empirically.

5 *Evaluation and error analysis*

We have designed a protocol for the manual review of the extraction results by a lexicographer. Reviewing is done by evaluating the suitability of the candidates with respect to the WordNet sense taken into consideration. The evaluation of Portuguese results has been done in an exploratory mode without a preestablished error typology. After the elaboration of that typology, based on this previous evaluation of the Portuguese candidates, we have been able to register an error type for each bad candidate found during the evaluation of Galician results. Therefore, only in the case of Galician, erroneous candidates have also received a code that indicates the reason for their exclusion.

We have obtained 1,832 multi-word candidates for Galician and 12,172 for Portuguese. The reduced number of candidates, when compared with the total number of different multi-word expressions from English, is related to the corpora lexical variety. For instance, the English texts in the English–Galician corpus only contain 2,174 different multi-word expressions from the 60,073 included in the PWN list. At this moment, 500 candidates for Galician have been evaluated. The percentage of correct answers in the evaluated candidates reaches 73.2% of the cases. A similar number of candidates were evaluated for Portuguese, obtaining a 75.5% of correct answers in this case.

The difference in accuracy between the proposed approach and that of 85% reported in (Vintar and Fišer, 2008) can be attributed to two factors. On the one hand, Vintar and Fišer work with legal texts in the field of specialised terminology extraction, where accuracy tends to be higher than in general vocabulary acquisition. On the other hand, and although we cannot compare directly the score values as the similarity measures were computed by different algorithms, Vintar and Fišer use a threshold of 0.05 as the lowest possible similarity score, thus probably decreasing the coverage of their experiment.

At first our expectation was to have a high accuracy, given that multi-word terms are less ambiguous than single-word terms. Figures 2 and 3 show the ambiguity of single and multi-word terms. The X-axis is the number of synsets a variant belongs to, while the Y-axis is the number of variants. For example, there is a single-word variant belonging to 75 different synsets, while the most ambiguous multi-word variant just appears in 19 different synsets.

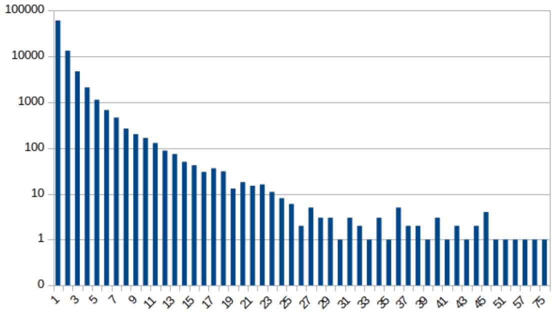


Figure 2: Ambiguity of single-word terms (number of variants vs number of synsets)

While these two images would confirm our expectation, some problems were found and erroneous candidates were generated for different reasons. In the following sections, we will describe and exemplify the most significant causes of error in the extraction process. Their incidence in the process of extracting Galician multi-word terms is shown in Table 2, where spelling errors are not considered, as explained below.

[9] $\mathcal{P}_f$ stands for the probability in the PTD computed from the forms corpus, while $\mathcal{P}_l$ is the probability from the lemmatised corpus.

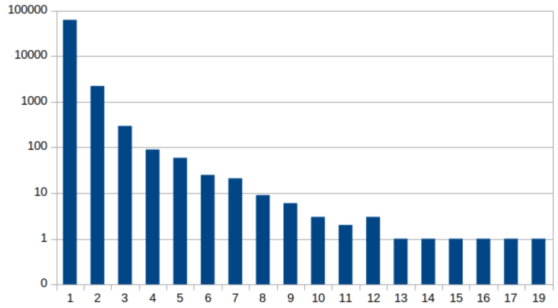


Figure 3: Ambiguity of multi-word terms (number of variants vs number of synsets)

Translation mistakes	18%
Idioms	26%
Transpositions	22%
Collocations	16%
Coordination	8%
Other types of error	10%

Table 2: Error typology in Galician multi-word term extraction

5.1 Spelling

In a few cases, the candidate generated from the corpus represents a variant rejected by the current official Galician normative. For example, the proposed candidate "*hospital siquiátrico*" for the concept of "*mental hospital*" (ili-30-03746574-n)[10] is not well written following the current regulations of the Galician language, which prescribe "*hospital psiquiátrico*" with initial "*p*" in the second word.

There are 8 errors of this type between the 500 candidates, from which 7 would be correct candidates with the corresponding spelling normativisation. These erroneous candidates cannot be considered as the result of any dysfunction of the extraction methodology, and can be easily identified and corrected during manual importation into the Galician wordnet, so they have not been taken into account for the evaluation of the accuracy of the results (and this is why they are not included in the data shown in Table 2).

5.2 Translation mistakes

Sometimes, the texts in the corpus contain translation mistakes that affect a multi-word in English and that lead to the generation of a wrong translation candidate.

For example, the English nominal compound "*numbers racket*", which has the meaning of "*an illegal daily lottery*" (ili-30-00508547-n),[11] is translated into Galician as "*raqueta de números*" (literally, "*racket of numbers*") in the OpenSubtitles2018 corpus, using the same words as in the source language, when the correct translation would be "*lotaría*" or "*lotaría ilegal*" depending on the context. In the same corpus, the English multi-word "*straight razor*" (ili-30-04332074-n)[12] has been rendered in Portuguese by "*gillete recta*" (literally, "*straight Gillette*"), when the correct translation would be "*navalha de barbear*", apart from the fact that Gillette is written with two t's. Therefore, these translation errors have led to the generation of the proposals "*gillete recta*" for Portuguese and "*raqueta de números*" for Galician which are bad translation candidates for their respective English terms.

5.3 Idioms

WordNet does not include free combinations, and some multi-word terms (like "*piece of cake*") have both an idiomatic sense (the one we found registered in WordNet) and the literal sense (not registered in WordNet).

Idioms are lexical sequences where the words mean something other than their literal meaning. In some cases, an erroneous candidate for an idiomatic multi-word is produced from its literal translation.

For instance, the erroneous proposal "*mina de sal*" for the meaning "*a job involving drudgery and confinement*" (ili-30-00606119-n)[13] is generated from the literal version of the English multiword "*salt mine*".

5.4 Transpositions

Shifts or transpositions in translation involve a change in the grammar from source language to target language. In some cases, the right translation from an English multi-word to a Galician or Portuguese term implies the change of the English noun group to a target language single noun.

This can cause errors in the application of the algorithm when it is possible to de-

[10] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-03746574-n`

[11] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-00508547-n`

[12] `http://wordnet.pt/synset/04332074-n`

[13] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-00606119-n`

tect incorrect lexical alignments in the parallel corpora possibly due to a bad translation, as in the incorrect translation proposal for Galician "*pezas de mobiliario*" for the English "*piece of furniture*" (ili-30-00606119-n),[14] where the correct translation would be "*moble*"; or as in the improper candidate for Portuguese "*criança macho*" for the English "*male child*" (ili-30-10285313-n),[15] where the proper Portuguese equivalent would be "*menino*".

5.5 Collocations

In lexicology, collocations are sequences of two or more words that usually go together, like "*strong tea*" or "*false teeth*". Collocations are highly idiomatic and ruled by the norms of language use.

Sometimes extraction produces results that are grammatically and semantically correct in the target language, but do not follow its rules of use. For example, the translation proposed "*traballo de policía*" for "*police work*" (ili-30-00606119-n)[16] is incorrect in Galician, because usage prescribes "*traballo policial*" for this concept, with the adjective "*policial*" instead of the prepositional phrase "*de policía*". The same kind of error in extraction can be appreciated in the Portuguese candidate "*água santa*" for English "*holy water*" (ili-30-14846517-n),[17] where the language rules of use would prescribe "*água benta*" (literally, "*blessed water*").

5.6 Coordination

Extraction rules may fail when applied to coordinated structures. For example, when processing the EN–GL alignment from the OpenSubtitles2018 corpus:

> EN: *Iraq has chemical and biological weapons which could be activated within 45 minutes*
> GL: *Iraq posúe armas químicas e biolóxicas que poderían ser activadas en menos de 45 minutos*

the extraction algorithm proposes the equivalence between the original "*biological weapon*" and the translation "*armas químicas*" ("*chemical weapons*"), where the proper Galician equivalent would be "*armas biolóxicas*" ("*biological weapons*").

5.7 Other types of errors

Other types of errors occur with a lower level of significance. For instance, a possible cause of error, which occurs only twice for Galician in our evaluation, is the lexical ambiguity of the English multi-word in WordNet. In this case, the extraction process is applied to all the senses of the term, with a high risk of error. The lexical form "*sea horse*", for example, has two senses in the English WordNet: the first with the meaning of "*walrus*" (ili-30-02081571-n)[18] and the second with the meaning of "*small fish with horse like heads bent sharply downward and curled tails*" (ili-30-01456756-n)[19]. Because of this, the extraction algorithm proposes the incorrect equivalence between the English "*sea horse*" (ili-30-01456756-n) with the sense of small fish and the Galician "*morsa*" ("*walrus*"), where the proper Galician term would be "*cabaliño de mar*" (literally, "*little horse of the sea*").

6 Conclusions and future work

In this paper we proposed a methodology to extract multi-word variant candidates in Portuguese and Galician using the original multi-word variants from the English WordNet aided by parallel corpora and translation patterns. Despite the difficulties, the results of human evaluation in section 5 show that the presented methodology, applied to the enlargement of wordnets with general vocabulary, leads to results not so different from those reported in previous works in the field of specialised terminology extraction (Gómez Guinovart and Simões, 2009). This would demonstrate the importance of associating morphology-based translation patterns to lexical alignment for the identification of multi-word WordNet variant candidates in parallel corpora.

Although the obtained accuracy is reasonable, better results could be achieved using higher threshold values. Looking to the Portuguese language results, for instance, the average score for wrong variants is 0.2172, while for the correct variants is 0.2589. Even

[14] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-03405725-n`

[15] `http://wordnet.pt/synset/10285313-n`

[16] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-00635012-n`

[17] `http://wordnet.pt/synset/14846517-n`

[18] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-02081571-n`

[19] `http://sli.uvigo.gal/galnet/galnet_var.php?ili=ili-30-01456756-n`

though the values are quite near, they show that it might be possible to obtain better accuracy values. Nevertheless, unlike (Vintar and Fišer, 2008), and given that all variants to be imported in Galnet and PULO will be manually validated, we preferred not to raise the threshold of the lowest possible similarity score and to obtain a bigger coverage of WordNet multi-word expressions.

The linguistic kinship between Portuguese and Galician has allowed us to apply the same techniques to carry out the task proposed in this research, and the results obtained for each language have been similar.

Future work includes both finishing the validation of the full set of Galician and Portuguese extracted variants, and introducing the validated variants in the Galnet and PULO knowledge databases.

7 Acknowledgements

This research has been carried out thanks to the project DeepReading (RTI2018-096846-B-C21) supported by the Ministry of Science, Innovation and Universities of the Spanish Government and the European Fund for Regional Development (MCIU/AEI/FEDER), and was partially funded by Portuguese National funds (PIDDAC), through the FCT – Fundação para a Ciência e Tecnologia and FCT/MCTES under the scope of the project UIDB/05549/2020.

References

Gómez Guinovart, X. and A. Oliver. 2014. Methodology and evaluation of the Galician WordNet expansion with the WN-Toolkit. *Procesamiento del Lenguaje Natural*, 53:43–50.

Gómez Guinovart, X. and A. Simões. 2009. Terminology extraction from English-Portuguese and English-Galician parallel corpora based on probabilistic translation dictionaries and bilingual syntactic patterns. In *Proc. of the Iberian SLTech 2009*, pages 13–16.

Gómez Guinovart, X. and M. A. Solla Portela. 2018. Building the Galician wordnet: Methods and applications. *Language Resources and Evaluation*, 52(1):317–339.

González-Agirre, A. and G. Rigau. 2013. Construcción de una base de conocimiento léxico multilingüe de amplia cobertura: Multilingual Central Repository. *Linguamática*, 5(1):13–28.

Lloberes, M., A. Oliver, S. Climent, and I. Castellón. 2013. Tratamiento de multipalabras en WordNet 3.0. In A. L. et al., editor, *Applied Linguistics in the Age of Globalization*. Universitat de Lleida, Lleida, pages 141–158.

Miller, G. A., R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. 1990. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3:235–244.

Oliver, A. 2014. WN-Toolkit: Automatic generation of wordnets following the expand model. In *Proc. of the 7th Global WordNet Conference*, pages 7–15, Tartu. GWN.

Padró, L. and E. Stanilovsky. 2012. Freeling 3.0: Towards wider multilinguality. In *Proc. of the Eight International Conference on Language Resources and Evaluation*, pages 2473–2479, Istanbul. ELRA.

Simões, A. and X. Gómez Guinovart. 2018. Extending the Galician wordnet using a multilingual Bible through lexical alignment and semantic annotation. volume 62 of *OpenAccess Series in Informatics*, pages 14:1–14:13, Dagstuhl. Schloss Dagstuhl.

Simões, A. and J. J. Almeida. 2003. NATools: A statistical word aligner workbench. *Procesamiento del Lenguaje Natural*, 31:217–224.

Simões, A. and X. Gómez Guinovart. 2013. Dictionary alignment by rewrite-based entry translation. In J. P. Leal, R. Rocha, and A. Simões, editors, *2nd Symposium on Languages, Applications and Technologies*, volume 29 of *OpenAccess Series in Informatics*, pages 237–247, Dagstuhl. Schloss Dagstuhl.

Vintar, S. and D. Fišer. 2008. Harvesting multi-word expressions from parallel corpora. In *Proc. of the 6th International Conference on Language Resources and Evaluation*, pages 1091–1096, Marrakech. ELRA.

Vossen, P., editor. 1998. *EuroWordNet: A multilingual database with lexical semantic networks*. Kluwer, Norwell.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 53-60 recibido 21-11-2019 revisado 07-01-2020 aceptado 09-01-2020

Metaphoricity Detection in Adjective-Noun Pairs

Detección de Metaforicidad en Pares Adjetivo-Sustantivo

Andrés Torres Rivera, Antoni Oliver, Marta Coll-Florit
Universitat Oberta de Catalunya
{atorresrive, aoliverg, mcollfl}@uoc.edu

Abstract: In this paper we propose a neural network approach to detect the metaphoricity of Adjective-Noun pairs using pre-trained word embeddings and word similarity using dot product. We found that metaphorical word pairs tend to have a lower dot product score while literal pairs a higher score. On this basis, we compared seven optimizers and two activation functions, from which the best performing pairs obtained an accuracy score of 97.69% and 97.74%, which represents an improvement of 6% over other current approaches.
Keywords: NLP, Metaphor, Word Embeddings, Deep Learning

Resumen: En este artículo proponemos un acercamiento mediante redes neuronales para la detección de la metaforicidad de pares Adjetivo-Sustantivo empleando *word embeddings* pre-entrenados y similitud de palabras mediante el producto escalar. Encontramos que los pares de palabras metafóricos tienden a tener un producto escalar bajo mientras que los pares no metafóricos un resultado más alto. Bajo este supuesto, comparamos siete optimizadores y dos funciones de activación, de las cuales los pares con mejor desempeño obtuvieron una exactitud de 97.69% y 97.74%, que representa una mejora de 6% sobre otros enfoques actuales.
Palabras clave: PLN, Metáfora, Word Embeddings, Aprendizaje Profundo

1 *Introduction*

The automatic detection of figurative language is one of the most challenging tasks in Natural Language Processing (NLP). Specifically, metaphor is the most studied process, as it is omnipresent in natural language text and therefore it is crucial in automatic text understanding (Shutova, 2010).

According to the Conceptual Metaphor Theory (Lakoff and Johnson, 1980), a metaphor represents a mapping of abstract concepts (target domain) to more concrete or tangible phenomena (source domain), as in the following examples, which are instances of the conceptual metaphor TIME IS MONEY:

You're wasting my time.

This gadget will save you hours.

Two main kinds of metaphor can be distinguished: *conventional* metaphors, which are commonly used in everyday language (as the examples above), and *novel*, *literary*, *creative* or *unconventional* metaphors, which surprise our imagination.

The study of metaphor is a prolific area of research in Cognitive Linguistics, being the Metaphor Identification Procedure (MIP) (Pragglejaz Group, 2007) and its derivative MIPVU (Steen et al., 2010) the most standard methods for manual metaphor detection. Moreover, in the area of Corpus Linguistics, some methods have been developed for annotation of metaphor in corpora (Shutova, 2017; Coll-Florit and Climent, 2019).

In reference to NLP, methodologies for automatic processing of metaphors can be classified into three main categories (Veale, Shutova, and Klebanov, 2016):

- *Corrective approaches*, the earliest ones, where metaphors are considered as a deviation of literal language that must be corrected.
- *Analogical approaches* where metaphors are viewed as some cross-domain transfer of semantic structure.
- *Schematic approaches* where each metaphorical expression is understood as an instance of a more general metaphorical schema.

ISSN 1135-5948. DOI 10.26342/2020-64-6

All these approaches have the following points in common: (1) assume the existence of a literal (or at least normative) meaning of words; (2) assume that some form of structural mapping is required to obtain an interpretation of the metaphor; and (3) assume that metaphor itself is a unit of conceptual representation.

According to Shutova (2010), there are two main tasks in the automatic processing of metaphors:

- *Metaphor recognition*: distinguishing between literal and metaphorical language in a text.
- *Metaphor interpretation*: identifying the intended literal meaning of a metaphorical expression.

Recently, techniques for metaphor recognition are shifting from classical machine learning techniques, as classifiers and decision trees, to the use of more advanced Artificial Intelligence techniques, as neural networks.

The main goal of this paper is to present a new model for metaphor recognition, and specifically for metaphoricity detection of adjective-noun pairs, from a neural network approach. Below we describe the main related works (section 2). Next we present our methodology and model (section 3) and the main results (section 4). We finish with the discussion and our overall conclusions (sections 5 and 6).

2 Related work

Current approaches regarding metaphor recognition include the works of Rosen (2018), Wu et al. (2018) and Mu, Yannakoudakis, and Shutova (2019), which focus on the detection of metaphorical instances in general corpora. Our work focuses on a different task within the scope of metaphor recognition that consists on detecting the metaphoricity of adjective-noun (AN) pairs in English as isolated units. Current approaches on this task include the works by Turney et al. (2011), Tsvetkov et al. (2014), Gutierrez et al. (2016) and Bizzoni, Chatzikyriakidis, and Ghanimifard (2017).

In relation to metaphor recognition in general corpora, Rosen (2018) developed an algorithm using deep learning techniques that uses a representation of metaphorical constructions in an argument - structure level. The algorithm allows for the identification of source-level mappings of metaphors. The author concludes that the use of deep learning algorithms including construction grammatical relations in the feature set improves the accuracy of the prediction of metaphorical source domains.

Wu et al. (2018) propose to use a Convolutional Neural Network - Long-Short Term Memory (CNN-LSTM) with a Conditional Random Field (CRF) or Softmax layer for metaphor detection in texts. They combine CNN and LSTM to capture both local and long-distance contextual information to represent the input sentences.

Some authors (Mu, Yannakoudakis, and Shutova, 2019) argue that using broader discourse features can have a substantial positive impact for the task of metaphorical identification. They obtain significant results using document embeddings methods to represent an utterance and its surrounding discourse. With this material a simple gradient boosting classifier is trained.

With regard to metaphoricity detection in AN pairs, the work of Turney et al. (2011) is based on the hypothesis that metaphorical word usage is correlated with the degree of abstractness of the context of a word. The idea comes from research in Cognitive Linguistics that views metaphor as a cognitive strategy to map knowledge between two domains: one of the domains is familiar, well-understood or concrete; and the other domain is unfamiliar, less understood or more abstract. They present an algorithm to classify a word sense in a given context as literal (denotative) or metaphorical (connotative) and evaluate the algorithm in a set of annotated AN phrases. One of the strengths of the approach is that it can generalize to new words outside the training data.

In Tsvetkov et al. (2014) a model to discriminate whether a syntactic construction has a literal or metaphoric sense is presented. The model uses lexical semantic features of the words in the construction. One of the advantages of the model is that it can be transferred to other languages by pivoting through a bilingual dictionary. They work with two syntactic constructions: subject-verb-object (SVO) and, like in our study, adjective-noun (AN) tuples.

In Gutierrez et al. (2016) a test case for compositional distributional semantic models (CDSMs) is presented. The authors propose a method to learn metaphors as lineal transformations in a vector space. They show that modeling metaphor explicitly within a CDSM can improve the resulting vector representations. As metaphors show a high degree of systematicity, it is possible to learn linear transformations for the representation of metaphorical mappings for adjetives in the same semantic domain.

Finally, in Bizzoni, Chatzikyriakidis, and Ghanimifard (2017) a single neural network with pre-trained vector embeddings is used to identify metaphors in AN pairs. The system is able to provide a metaphoricity score as an output. Table 1 presents the accuracy score of the current approaches in AN metaphoricity detection which establishes a current performance of 91% in accuracy.

The approaches proposed by Turney et al. (2011) and Tsvetkov et al. (2014) implement feature engineering (FE) using small annotated (Ann.) datasets. Currently, Gutierrez et al. (2016) and Bizzoni, Chatzikyriakidis, and Ghanimifard (2017) opt for approaches that do not implement FE, instead both present models trained using embeddings: a distributional semantic model (DSM) in the first case, and word2vec in the second case. In both instances the training and testing data was generated using the AN corpus compiled by Gutierrez et al. (2016).

	A	**FE**	**Ann.**
Turney et al. (2011)	0.79	Y	100
Tsvetkov et al. (2014)	0.85	Y	200
Gutierrez et al. (2016)	0.81	N	8592
Bizzoni et al. (2017)	0.91	N	8592

Table 1: Accuracy score comparison of metaphorical AN pairs detection

3 Methodology

Our study is based on the annotated AN pairs corpus presented by Gutierrez et al. (2016), which is composed by 8,592 word pairs that are a combination of 23 unique adjectives and 3,418 unique nouns. This corpus can be divided in two subsets: one composed by 4,601 metaphoric pairs and another composed by 3,991 literal or non metaphorical pairs with an interannotator reliability of $\kappa = 0.80$ and a standard error (SE) of 0.2. Both subsets include cases of metaphoric pairs with each of the 23 adjectives, but in the case of nouns the metaphoric subset contains 2,027 unique nouns whereas the non metaphoric subset 1,547 unique nouns.

Gutierrez et al. (2016) focused on adjectives that function as source domain words in productive conceptual metaphors (CM). Some examples of this kind of CM found in the AN corpus include: *bright day*, *rough character*, *heavy expansion*, and *bitter competition*. As shown in Table 2, the 23 adjectives were divided in eight source-domain categories.

Source	**Adjectives**
Temperature	Cold, heated, icy, warm
Light	Bright, brilliant, dim
Texture	Rough, smooth, soft
Substance	Dense, heavy, solid
Clarity	Clean, clear, murky
Taste	Bitter, sour, sweet
Strength	Strong, weak
Depth	Deep, shallow

Table 2: Categories of the 23 adjectives that compose the AN corpus

We used pre-trained word vectors that were trained using part of the Google News dataset. This model contains 300-dimensional vectors with a context window size of 5 (Mikolov et al., 2013a; Mikolov et al., 2013b; Mikolov, Yih, and Zweig, 2013; Le and Mikolov, 2014). We opted to use these vectors in order to reproduce the process followed by Bizzoni, Chatzikyriakidis, and Ghanimifard (2017).

3.1 Dot product as a similarity measure

Within an Euclidean space, the dot product (Equation 1) is the result of multiplying the magnitudes of two equal-length vectors and the cosine of the angle between them. The result of this operation is a scalar value that can be interpreted as the similarity between to vectors: vectors that have a low score tend to be less similar while vectors that have a higher score tend to be more similar. Word embeddings are n-dimensional vectors that contain semantic and lexical information from all the words that compose the training vocabulary. Computing the dot product between two given word vectors might indicate the similarity relation that exists between them, as shown by Mikolov, Yih, and Zweig (2013), inasmuch as similar words tend to appear near each other within a vector space.

$$\mathrm{A} \cdot \mathrm{B} = ||\mathrm{A}||\ ||\mathrm{B}|| \cos\theta \tag{1}$$

After computing the dot product for each AN pair, we observed that metaphorical pairs presented a mean result of 0.8548 with an standard deviation (SD) of 0.6865, while the mean result for literal pairs was 1.2545 with a SD of 0.8418. As shown in Figure 1, metaphorical AN pairs (blue) tend to have a lower dot product score while literal AN pairs (orange) have a higher score, which might indicate that literal AN pairs tend to be more similar, and metaphorical AN pairs are combinations of words that are less similar.

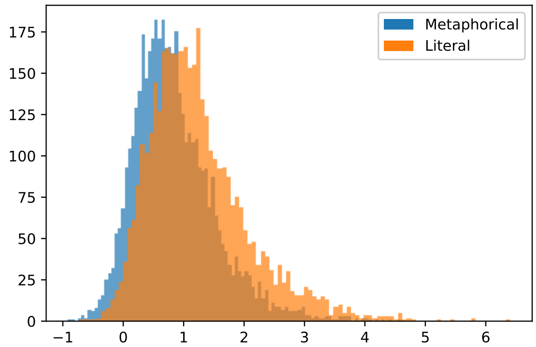


Figure 1: Dot product comparison of metaphorical and literal AN pairs

This values are observed across all sources, Table 3 shows the mean dot product score by source and tag (Literal or Metaphoric). Only the source *Strength* has a higher metaphoric mean dot product in comparison with its literal counterpart. In all the other sources the literal AN pairs have a higher mean dot product result. In some cases such as the sources *Depth* and *Texture* this score almost doubles the value obtained by the metaphoric AN pairs.

The five highest dot product scores were obtained by literal AN pairs belong to the *Temperature* source, such as: *icy snow, icy arctic, icy blizzard, cold child* and *icy precipitation.* On the other hand, the lowest dot product score where mostly obtained by metaphorical AN pairs that belong to different sources, such as: *clean datum, bitter identification, brilliant parent* and *rough customer*, with one instance of the literal pair *shallow outfit.*

Source	**Tag**	**Mean**	**SD**
Clarity	Lit.	0.957033	0.705107
	Met.	0.733197	0.544971
Depth	Lit.	1.564873	0.865550
	Met.	0.778630	0.553173
Light	Lit.	1.276814	0.859143
	Met.	0.824224	0.647932
Strength	Lit.	0.628803	0.433746
	Met.	0.799933	0.583103
Substance	Lit.	1.019069	0.592838
	Met.	0.650521	0.548152
Taste	Lit.	1.996791	0.884432
	Met.	1.270854	0.887818
Temperature	Lit.	1.352197	0.938974
	Met.	0.993028	0.770670
Texture	Lit.	1.209835	0.585611
	Met.	0.699859	0.580966

Table 3: Mean dot product score and standard deviation (SD) by source and tag

3.2 Model description

Our model consists of a variation of the first architecture proposed by Bizzoni, Chatzikyriakidis, and Ghanimifard (2017). Under this architecture, a network is a generalization of the additive composition model (Equations 2 and 3) proposed by Mitchell and Lapata (2010), but using a weight matrix W that modifies all feature dimensions at the same time.

$$\mathbf{p} = (\mathbf{u}, \mathbf{v}; \theta) \tag{2}$$

$$\mathbf{p} = W_{adj}^{T}\mathbf{u} + W_{noun}^{T}\mathbf{v} + b \tag{3}$$

This approach can be implemented by concatenating word vectors before feeding them to a neural network. In this case, the parameter function is defined according to equations (4) and (5):

$$W = \begin{bmatrix} W_{adj} \\ W_{noun} \end{bmatrix} \tag{4}$$

$$\mathbf{p} = f_{\theta}(\mathbf{u}, \mathbf{v}) = W^{T} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} + b \tag{5}$$

Using the observed scores of the dot products of the AN pairs, we propose a variation of the multiplicative model presented by Mitchell and Lapata (2010), where instead of computing tensor multiplication we compute the dot product of each AN pair using their embeddings. With this modification we obtain the projection of vector **u** over **v** (Equation 6), and thus the network is fed a scalar value that can be interpreted as the similarity relation that exists between a given word vector pair.

$$\mathbf{p} = f_\theta(\mathbf{u}, \mathbf{v}) = W_{adj}^T \mathbf{u} \cdot W_{noun}^T \mathbf{v} + b \qquad (6)$$

To evaluate the performance of our model we compared the accuracy score of 7 optimizers (Adam, Nadam, Adamax, Adagrad, Adadelta, Stochastic Gradient Descent [SGD] and RMS Prop) with ReLu and linear function as activation functions. In all cases we set binary cross-entropy as the loss function, and used a 10 K-fold cross validation to obtain the mean accuracy score of each optimizer-activation pair. After performing this evaluation we proceeded to evaluate the best performing models to compare their mean accuracy error, precision, recall and f1-score. The model was trained using the same parameters proposed by Bizzoni, Chatzikyriakidis, and Ghanimifard (2017), i.e. it was trained for 20 epochs using 500 examples for training and the rest for testing.

4 Results

After training each model we calculated the mean accuracy using 10 K-Fold cross validation. As shown in Table 4, the set of optimizers using the linear activation function obtained a mean of 97% accuracy. The highest score was obtained by the model trained using the Adagrad optimizer, which obtained an accuracy equal to 97.69%, while the lowest scoring model was the one trained using SGD with an accuracy equal to 69.97%.

Optimizer	A	SD
Adam	97.58	0.4622
Nadam	97.62	0.5270
Adamax	97.56	0.5799
Adagrad	**97.69**	0.5182
Adadelta	97.65	0.5798
SGD	69.49*	15.0106
RMS Prop	97.48	0.3358

Table 4: Linear Function Accuracy Score (A) and Standard Deviation (SD)

The second set of optimizers was trained using ReLu as activation function. In this case the overall scores where around 97%, the highest accuracy score was 97.74%, obtained by the model trained using Nadam+ReLu. We can also observe a considerable improvement in the case of SGD+ReLu, which obtained an accuracy score of 92.16%. This represents an improvement of 22.67% in comparison with its SGD+linear function equivalent.

Optimizer	A	SD
Adam	97.63	0.4012
Nadam	**97.74**	0.4475
Adamax	97.44	0.5505
Adagrad	97.61	0.4909
Adadelta	97.51	0.4429
SGD	92.16*	3.8450
RMS Prop	97.52	0.4888

Table 5: ReLu Accuracy Score (A) and Standard Deviation (SD)

Overall, we can observe an improvement of 6% over the 91% of the current approach. Nevertheless, using accuracy as the only evaluation metric can lead to misinterpretations since an increase in accuracy might not indicate an increase in predictive ability. To ensure that the increase in accuracy of this methodology corresponds to an increase in performance, we proceeded to compare the two optimizer+activation pairs that had the highest accuracy score (Adagrad+Linear function, and Nadam+ReLu) using precision, recall and f1-score, in order to ensure that the models are capable of generalization.

Opt.	MAE	P	R	F1
Adagrad	0.0305	**0.9675**	**0.9829**	**0.9751**
Nadam	0.0325	0.9645	0.9785	0.9714

Table 6: Mean Absolute Error (MAE), Precision (P), Recall (R) and f1-Score (F1) results of the Adagrad and Nadam Optimizers

In Table 6 it can be observed that the Adagrad+Linear function model had better performance than the Nadam+ReLu model in all cases that were evaluated, mainly in recall where the Adagrad+Linear function model obtained 98.29%. In the case of the f1-metric, the Adagrad+Linear function model had better performance by a margin of 0.37%. Nevertheless, both models present a significant improvement over the current state of the art.

5 Discussion

The multiplicative models presented by Mitchell and Lapata (2010) operate using tensor multiplication or word vector cross products. While Bizzoni, Chatzikyriakidis, and Ghanimifard (2017) analyzed the performance of a multiplicative approach, this operation might have created a new vector or representation that lost the lexical information provided by the embeddings, and therefore the performance of the model.

Vector concatenation maintains the sequence and order of the AN pairs that are being feed to the network, but it does not take into account their lexical or semantic relationships. While the dot product of word vectors loses the word order, this measure can interpret the similarity between the word pair that is being analyzed. Moreover, since all the AN pairs follow the same structure, in this context word order or word vector order might be of less importance than the semantic relation between them.

A scalar value reduces the dimensionality of the input from $W \in \mathbb{R}^{300\times600}$ and $b \in \mathbb{R}^{300}$ to a single scalar value $W \in \mathbb{R}^{300\times1}$, thus producing a simpler model with a single feature created based on the word vectors of each component of each AN pair. In our case the metaphoricity vector interprets this scalar value as the lexical-semantic relation between each pair and obtains a representation that determines its metaphoricity.

Regarding the dot product scores of the source *Strength*, we used a t-distributed stochastic neighbor embedding (t-SNE) initialized with principal component analysis (PCA) to reduce the dimensionality of the embeddings from 300 to 2 to visualize the adjectives and their pairing nouns. In Figure 2 it can be observed that nouns ("x"[1]) seem to cluster in the center of the vector space along with both *Strength* adjectives ("triangle").

When performing the same analysis with other sources such as *Depth*[2] (Figure 3), the plot shows that nouns tend to be distributed throughout the vector space in a more sparse manner, which could explain why in the case of *Strength* metaphorical AN pairs tend to have a higher dot product mean.

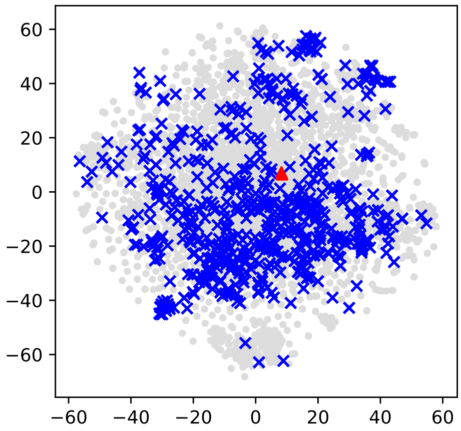


Figure 2: Strength t-SNE-PCA plot

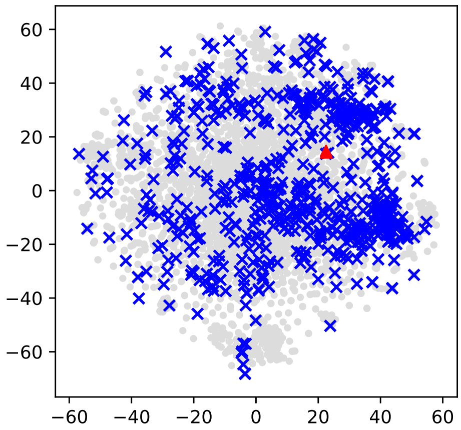


Figure 3: Depth t-SNE-PCA plot

6 Conclusion

In this paper we have presented an approach to AN metaphor detection by implementing a fully connected neural network using pretrained word embeddings. Our multiplicative model consists in computing the dot product between the word vectors of each of the components of the AN pair that is fed to the network. By reducing the dimensionality of the input parameter, this approach introduces a simpler approach to AN metaphor detection while improving the performance of the model.

We evaluated seven optimizers paired with two different activation functions, and in most cases every combination obtained a higher accuracy score in comparison with the current state of the art: an overall of 97% accuracy which represents an improvement of 6% over the 91% reported by Bizzoni, Chatzikyriakidis, and Ghanimifard (2017). To further asses our results we evaluated the

[1] The gray dots represent all the remaining nouns and adjectives of the vocabulary.

[2] We have chosen Depth because it has a similar number of unique nouns (638) as Strength (625).

top performing models using precision, recall, and f1-score which was not reported in the related works.

Both models obtained 97% in f1-score, and more precisely –after validating the results using 10 K-fold cross validation– the Adagrad+Linear function model obtained 97.51% and the Nadam+ReLu 97.14%. In each instance the only training data where the pre-trained Google News word2vec embeddings, no other features were used during the training process.

Acknowledgements

This research was conducted in the framework of the project *MOMENT: Metaphors of severe mental disorders. Discourse analysis of affected people and mental health professionals*, funded by the Spanish National Research Agency (Agencia Estatal de Investigación, AEI) and the European Regional Development Fund, within the National Programme for Research Aimed at the Challenges of Society. Ref. FFI2017-86969-R (AEI/FEDER, UE)

References

Bizzoni, Y., S. Chatzikyriakidis, and M. Ghanimifard. 2017. "Deep" Learning : Detecting Metaphoricity in Adjective-Noun Pairs. In *Proceedings of the Workshop on Stylistic Variation*, pages 43–52, Copenhagen, Denmark. Association for Computational Linguistics.

Coll-Florit, M. and S. Climent. 2019. A new methodology for conceptual metaphor detection and formulation in corpora. a case study on a mental health corpus. *SKY Journal of Linguistics*, 32.

Gutierrez, E., E. Shutova, T. Marghetis, and B. Bergen. 2016. Literal and Metaphorical Senses in Compositional Distributional Semantic Models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 183–193, Berlin, Germany. Association for Computational Linguistics.

Lakoff, G. and M. Johnson. 1980. *Metaphors we Live by.* University of Chicago Press, Chicago.

Le, Q. V. and T. Mikolov. 2014. Distributed Representations of Sentences and Documents. *arXiv:1405.4053 [cs]*, May. arXiv: 1405.4053.

Mikolov, T., K. Chen, G. Corrado, and J. Dean. 2013a. Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013b. Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of the 26th International Conference on Neural Information Processing Systems*, 2:9.

Mikolov, T., W. Yih, and G. Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751. Association for Computational Linguistics. event-place: Atlanta, Georgia.

Mitchell, J. and M. Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

Mu, J., H. Yannakoudakis, and E. Shutova. 2019. Learning Outside the Box: Discourse-level Features Improve Metaphor Identification. *arXiv:1904.02246 [cs]*, April. arXiv: 1904.02246.

Pragglejaz Group. 2007. MIP: A method for identifying metaphorically used words in discourse. *Metaphor and Symbol*, 22(1):1–39.

Rosen, Z. 2018. Computationally Constructed Concepts: A Machine Learning Approach to Metaphor Interpretation Using Usage-Based Construction Grammatical Cues. In *Proceedings of the Workshop on Figurative Language Processing*, pages 102–109, New Orleans, Louisiana. Association for Computational Linguistics.

Shutova, E. 2010. Models of Metaphor in NLP. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 688–697.

Shutova, E., 2017. *Annotation of Linguistic and Conceptual Metaphor*, pages 1073–1100. Springer Netherlands, Dordrecht.

Steen, G. J., A. G. Dorst, J. B. Herrmann, A. Kaal, T. Krennmayr, and T. Pasma. 2010. *A Method for Linguistic Metaphor Identification: From MIP to MIPVU*. John Benjamins.

Tsvetkov, Y., L. Boytsov, A. Gershman, E. Nyberg, and C. Dyer. 2014. Metaphor Detection with Cross-Lingual Model Transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 248–258, Baltimore, Maryland. Association for Computational Linguistics.

Turney, P., Y. Neuman, D. Assaf, and Y. Cohen. 2011. Literal and Metaphorical Sense Identification through Concrete and Abstract Context. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 680–690, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Veale, T., E. Shutova, and B. B. Klebanov. 2016. Metaphor: A Computational Perspective. *Synthesis Lectures on Human Language Technologies*, 9(1):1–160, February.

Wu, C., F. Wu, Y. Chen, S. Wu, Z. Yuan, and Y. Huang. 2018. Neural Metaphor Detecting with CNN-LSTM Model. In *Proceedings of the Workshop on Figurative Language Processing*, pages 110–114, New Orleans, Louisiana. Association for Computational Linguistics.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 61-68 recibido 22-11-2019 revisado 10-01-2020 aceptado 15-01-2020

AzterTest: Open Source Linguistic and Stylistic Analysis Tool

AzterTest: Herramienta de Análisis Lingüístico y Estilístico de Código Abierto

Kepa Bengoetxea, Itziar Gonzalez-Dios, Amaia Aguirregoitia
University of the Basque Country (UPV/EHU)
{kepa.bengoetxea,itziar.gonzalezd,amaia.aguirregoitia}@ehu.eus

Abstract: Text Analysis is a useful process to assist teachers in the selection of the most suitable texts for their students. This task demands the analysis of several text features, which is done mostly manually (e.g. syntactic complexity, words variety, etc.). In this paper, we present an open source tool useful for linguistic and stylistic analysis, called AzterTest. AzterTest calculates 153 features and obtains 90.09 % in accuracy when classifying into three reading levels (elementary, intermediate, and advanced). AzterTest is available also as web tool.
Keywords: text analysis, readability assessment, web application

Resumen: El análisis de texto es un procedimiento útil para ayudar a los profesionales de la educación en la selección de los textos más adecuados para sus alumnos. Esta tarea exige el análisis de varias características de texto (por ejemplo, complejidad sintáctica, variedad de palabras, etc.), que se realiza principalmente de forma manual. En este artículo, presentamos AzterTest, una herramienta de código abierto para el análisis lingüístico y estilístico. AzterTest calcula 153 características y obtiene una exactitud de 90.09 % al distinguir tres niveles de lectura (elemental, intermedio y avanzado). AzterTest también se encuentra disponible como herramienta web.
Palabras clave: análisis de texto, lecturabilidad, aplicación web

1 *Introduction*

According to the results of PISA 2015 (OECD, 2016), around % 20 of students in the OECD countries do not attain the baseline level of proficiency in reading. Reading is one of the most effective mechanisms in the learning process, since it is our reading capability what let us access and interpret all the available information.

Classifying and adapting reading educational contents manually for students with special needs is an expensive and a hard task for teachers and educators. However, language technologies and Natural Languages Processing (NLP) tools can play a relevant role in the classification and adaptation of available resources, as proved in the works presented in the series of BEA workshops organized by ACL SIGEDU[1]. These technologies have been proved to ease the burden of the educational professionals when selecting texts with a specific level and specific characteristics.

In this paper we present AzterTest, an open-source NLP based tool and web service. AzterTest analyzes 153 linguistic and stylistic features of texts, such as word frequency, sentence length, vocabulary level, argument overlap or use of connective devices. The aim of AzterTest is to provide a detailed linguistic and stylistic analysis of the text to help teachers find the most appropriate reading materials. To evaluate our tool, we test AzterTest in a readability assessment scenario and we compare AzterTest to Coh-Metrix (Graesser, McNamara, and Kulikowich, 2011), a well-known computational tool which analyzes the linguistic and discourse indices. However, its use is not limited to readability assessment, since it can also be used for other purposes such as stylometry or to assess differences among genres or varieties.

This paper is structured as follows: in Section 2 we introduce the related work, in Section 3 we present AzterTest, which we evaluate in Section 4. Later, we describe the web version in Section 5. Finally, we conclude and outline the future work in Section 6.

2 *Related work*

Traditionally, reading materials have been assessed with conventional readability formulae such as Flesch (Flesch, 1948), Dale-Chall

[1] `https://sig-edu.org/`

ISSN 1135-5948. DOI 10.26342/2020-64-7

(Chall and Dale, 1995), the indexes Gunning FOG (Gunning, 1968) or Simple Measure Of Gobbledygook (SMOG) grade (Mc Laughlin, 1969). In general, these formulae are based on raw features such as word and sentence length, vocabulary lists and frequencies and give a score to classify texts. NLP based tools have proved that these formulae are not reliable when assessing the levels of the texts (Si and Callan, 2001; Petersen and Ostendorf, 2009; Feng et al., 2010). Moreover, the information offered by these traditional formulae is insufficient, since they do not detect slight changes in aspects such as coherence and cohesion of the texts (Graesser, McNamara, and Kulikowich, 2011).

Computational tools for linguistic analysis, generally, focus on the quantitative dimension of text complexity, where features related to quantitative aspects of the texts (word length, frequency, incidence of grammar structures, etc.) are used to assess linguistic complexity. Concerning research for English, Coh-Metrix 3.0 (Graesser, McNamara, and Kulikowich, 2011) analyzes texts providing 110 measures in its free version, which are classified in 11 groups: descriptive, text easability principal components scores, referential cohesion, latent semantic analysis, lexical diversity, connectives, situation model, syntactic complexity, syntactic pattern density, word information and readability. Coh-Metrix[2] has also been partially adapted to Brazilian Portuguese (Scarton and Alusio, 2010) and Spanish (Quispersaravia et al., 2016). In the case of Spanish, the tool *El Manchador de Textos* analyses some linguistic features (Venegas, 2008).

These tools are mainly used for Readability assessment, which aims to grade the ease or the difficulty of written texts. Most of research in this line has focused on classifying texts as simple or complex e.g. for English (Feng et al., 2010), Italian (Dell'Orletta, Montemagni, and Venturi, 2011), German (Hancke, Vajjala, and Meurers, 2012), Spanish (Štajner and Saggion, 2013), or Basque (Gonzalez-Dios et al., 2014). However, only a few of them have also tried to assess three levels e.g. for Brazilian Portuguese (Aluísio et al., 2010), the CEFR levels for French (François and Fairon, 2012) or have developed a multilingual proposal (Madrazo and Pera, 2019). Besides, getting an open licensed corpora and data is the main problem when training these systems.

3 *AzterTest: Tools and Features*

In this section, we describe AzterTest which is freely available from a public GitHub repository[3] and is licensed under GNU General Public License v3.0. First of all, we outline the resources and tools used in the implementation process and later, we introduce the features computed by the tool.

3.1 Preprocessing tools and resources

AzterTest uses NLP-Cube (0.1.0.7) (Boroș, Dumitrescu, and Burtica, 2018) tool for automatic analysis, which is the state-of-the-art in segmentation (tokenization and sentence-splitting), lemmatization, POS tagging and dependency parsing task for over 50 languages. NLP-Cube was one of the best systems in English on CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies (Zeman and Hajič, 2018). We have also tested StanfordNLP (0.2.0) (Qi et al., 2019) because AzterTest is based on Universal Dependencies (UD) and it allows to use others UD parsers. Hence, AzterTest can be easily adapted to other languages in the near future.

Moreover, for the analysis of English text, we have implemented a syllable splitter based on CMUdict (Carnegie Mellon University Pronouncing Dictionary) (Weide, 2005). This splitter is used to count the number of syllables in a text.

To obtain lexical information we have used the stop words from the NLTK stopwords Corpus, the list of irregular verbs which are freely available[4], the Dale-Chall word list[5], a list of the 3,000 core words that every learner of English needs to know at A1-B2 level and an additional 2,000 word list including the most useful high-level words guiding advanced learners at B2-C1 level to learn to expand their vocabulary[6].

For word frequency, we have used wordfreq (Speer et al., 2018), which provides access to estimates of how often a word is used in 36 languages. wordfreq detects the word frequency

[2] The versions for Brazilian Portuguese and Spanish are based on Coh-Metrix 2.0 and the authors adapted around 40 Coh-Metrix indices related to cohesion, coherence and the difficulty of text comprehension, according to specific characteristics of each language.

[3] https://github.com/kepaxabier/AzterTest

[4] https://github.com/Bryan-Legend/babel-lang/blob/master/Babel.EnglishEmitter/Resources/Irregular

[5] http://www.readabilityformulas.com/articles/dale-chall-readability-word-list.php

[6] https://www.oxfordlearnersdictionaries.com/wordlists/oxford3000-5000

of a word as the logarithm in base 10 of the number of times a word appears per one billion words. A word rated as 3 appears 10^3 times for every 10^9 words, that is, once per million words. Using wordfreq and after testing different values, for our educational purposes, we have decided to consider words with a value below 4 as rare words.

For semantic information, we have used WordNet (Miller, 1995), which groups nouns, verbs, adjectives and adverbs into sets of cognitive synonyms (synsets), each expressing a distinct concept. Moreover, synsets are interlinked by means of conceptual-semantic and lexical relations.

For semantic similarity, we have used the Universal Sentence Encoder. The Universal Sentence Encoder encodes text into high dimensional vectors that can be used for text semantic similarity. The pre-trained Google's Universal Sentence Encoder (Cer et al., 2018) is publicly available in Tensorflow-hub[7].

3.2 Linguistic and Stylistic Features

Linguistic features are those related to morphology, syntax and semantics while the stylistic features are related to cohesion, vocabulary knowledge etc. In order to decide which features to implement in AzterTest, we have analyzed in detail the features provided by the works presented in Section 2. After a deep analysis, we have implemented a set of 153 AzterTest features. Following, we present the list of the features included in AzterTest organized by type:

- **Descriptive:** numbers and incidences of paragraphs, sentences, words, distinct words, words with punctuation; sentences per paragraph, words per sentence with and without stopwords; syllables per word, letters per lemma, and letters per word with and without stopwords.
- **Classical Readability formulae:** Flesch-Kincaid grade level, Flesch readability ease, Dale-Chall and SMOG.
- **Lexical Diversity:** lexical density, densities of nouns, verbs, adjectives and adverbs, simple type-token ratio, content type-token ratio, type-token ratio of nouns, verbs, adjectives and adverbs, lemma simple type-token ratio, lemma content type-token ratio, lemma noun, verb, adjective, and adverb type-token ratio, Honoré and Maas lexical density measures and Measure of Textual Lexical Diversity (MTLD).
- **Word Frequency:** words with a value below 4 in wordfreq as rare words.
- **Vocabulary knowledge:** numbers and incidences of A1, A2, B1, B2 and C1 level vocabulary in the text, and number and incidence of content words not in A1-C1 vocabulary.
- **Word Semantic information:** the average values of polysemy of words, hypernym values of verbs, hypernym values of nouns, hypernym values of nouns and verbs.
- **Word Morphological information:** incidences and numbers of nouns, adjectives, adverbs, pronouns and each type of pronoun (first, second and third person; singular and plural), verbs and all variations for verbs (tense, mood, regularity, etc).
- **Syntactic Complexity:** left embeddedness (mean of number of words before the main verb), means of the number of propositions, levels of dependency tree, subordinate clauses and relative subordinate clauses, verbs in gerund form, verbs in infinitive form; descendants and modifiers per noun phrase; verb and noun phrases per sentence, and punctuation marks per sentence.
- **Syntactic Pattern Density:** noun and verbal phrase density, passive voice agentless passive voice verbs density, negation density, infinitive and gerund form density.
- **Referential Cohesion:** noun, argument, stem and content word overlap, mean and standard deviation of semantic similarity.
- **Connectives (logical cohesion):** incidences of causal, logical, adversative/contrastive, temporal and conditional connectives.

The main contributions of this work to the field of feature analysis are the ones concerning word frequency and vocabulary knowledge. For a detailed explanation of the rationale behind each of these metrics we refer the interested reader to the AzterTest documentation[8].

[7] https://tfhub.dev/

[8] http://178.128.198.190/information.html

4 Extrinsic Evaluation: Readability Assessment of English Texts

In this Section we present an extrinsic evaluation of AzterTest in a readability assessment scenario for English texts. In this evaluation, we have tested various classifiers to detect three reading levels (elementary, intermediate, advanced) based on Coh-Metrix and AzterTest's output on an open licensed corpora. We compare our results to other systems, perform an error analysis and discuss the best features.

4.1 Corpus

In order to train and validate AzterTest, we have used the corpus OneStopEnglish corpus (Vajjala and Lucic, 2018). This corpus compiles newspaper articles aligned at text and sentence level across three reading levels (elementary, intermediate, advanced), targeting English as Second Language (ESL) learners. The corpus consists of 189 texts, each of them in three versions (567 in total). We have decided to use this corpus because it is one of the few available[9] and it is licensed under CC BY-SA 4.0. Moreover, this corpus demonstrates its usefulness for automatic readability assessment among others. Namely, Vajjala and Lucic (2018) obtained an accuracy of 78.13 % using features based on readability classification research with the Sequential Minimal Optimization (SMO) (Platt, 1998) classifier.

For our experimental purposes we have randomly divided the corpus (in total 567 texts) into 2 non-overlapping datasets: 456 texts (152 texts for each class) as the training set and 111 texts (37 texts for each class) as the test set.

4.2 Classifying Experiments and Results

In order to classify the texts according to their complexity level, we have trained several classifiers that are included in WEKA (Hall et al., 2009). To evaluate the classifiers we have used the 10-fold cross-validation and the test set.

In these experiments we have tested three tools: Coh-Metrix and two configurations of AzterTest. In the first configuration of AzterTest we have taken into account all the features (absolute numbers and ratios) while in the second, AzterTest-ratios, we have only selected features based on incidents, means or typical deviations. Regarding the features, first, we have tested all the Coh-Metrix, AzterTest and AzterTest-ratios features.

Secondly, in order to detect the best features to tag and automatically remove the noise ones, we have tested different sets of attributes (25, 50, 75 and 100). In this experiment, we have tested chi square using different sets of attributes: 25, 50, 75 and 100. We have used Flesch readability ease as baseline.

In Table 1 we present the accuracy of the classifiers (Class. column) for each tool [Baseline, Coh-Metrix (Coh), AzterTest (Azt) and AzterTest-ratios (Azt-r)] and using different features (Feat. column). For brevity, we only show the classifiers [*i*) Sequential Minimal Optimization (SMO) (Platt, 1998) and *ii*) Simple Logistic (SL) (Landwehr, Hall, and Frank, 2005)] and the feature sets (all, 50 and 25) that have obtained the best results. We have tested the classifiers with the defaults hyperparameters.

Tool	Class.	Feat.	Data	Accu.
Baseline	SMO	Flesch	Cross Test	49.78 54.95
Coh	SL	All	Cross Test	77.19 81.98
Coh	SL	50	Cross Test	**77.85** **81.98**
Coh	SL	25	Cross Test	75.65 85.58
Azt	SMO	All	Cross Test	82.01 84.68
Azt	SMO	50	Cross Test	**82.01** **90.09**
Azt	SMO	25	Cross Test	82.01 88.28
Azt-r	SMO	All	Cross Test	80.92 84.68
Azt-r	SMO	50	Cross Test	80.04 85.58
Azt-r	SMO	25	Cross Test	**81.35** **84.68**

Table 1: Classification results of the three level readability assessment experiment

Respecting the classification results, the best classifier for Coh-Metrix is Simple Logistic, while it is SMO for AzterTest's two configurations and the baseline. The best results are obtained with the 50 most predictive features in Coh-Metrix and AzterTest, but with 25 in AzterTest-ratios. Moreover, all the results are lower when evaluating with 10 fold cross-validation.

In sum, looking at these results, the best model is the SMO classifier with 50 features of

[9] https://zenodo.org/record/1219041

AzterTest, namely 90.09 %. It is 4.16 points better than the best Coh-Metrix results using the cross-validation and 8.11 points using the test set. AzterTest is also better than the baseline in 32.23 points with the cross-validation and in 35.14 points with the test set. Therefore, in this scenario, AzterTest outperforms Coh-Metrix, the classical readability formula used as baseline and the results reported by Vajjala and Lucic (2018). However, AzterTest-ratios is not far from AzterTest, and outperforms Coh-Metrix when evaluating with cross-validation.

In addition to the all and selected features, we have also trained the classifiers with each type of linguistic/stylistic features that we described in section 3.2. In Table 2 we rank these results by type.

Feature Group	Data	Accu.
Syntactic	Cross Test	70.61 75.67
Lexical Density	Cross Test	65.13 72.07
Descriptive	Cross Test	65.13 67.56
Word Frequency	Cross Test	60.96 68.46
Vocabulary	Cross Test	55.92 60.36
Readability	Cross Test	53.50 59.45
Word Morphological	Cross Test	50.21 55.85
Word Semantic	Cross Test	50.21 49.54
Referential Cohesion	Cross Test	46.05 44.14
Discourse Connectives	Cross Test	38.59 34.23

Table 2: The results of the SMO classifier with specific linguistic features (only AzterTest's ratios)

Taking into account the different feature types, the syntactic features (complexity and pattern density together) performed best with an accuracy of 70.61 %; lexical features and descriptive features (65.13 %) performed almost equally well; word frequencies performed worse than lexical in cross-validation but similarly in the test and, finally, the accuracy of referential cohesion and discourse connectives is below 50 %.

Finally, we present the F measure for each text level of the best model. The F measure is 0.917 for the elementary level, 0.857 for the intermediate and 0.932 for the advanced. Comparing these results to the work for Brazilian Portuguese (Aluísio et al., 2010) that also classified into three levels, rudimentary (F measure 0.732), basic (F measure 0.483) and advanced level (F measure 0.913), we observe that our model is stronger across classes.

4.3 Error analysis

We have also carried out an error analysis, using the output of the best system, which is the SMO classifier incorporating the best 50 characteristics of AzterTest. We have checked manually the annotation results in the test. The test set comprised 111 instances and only 11 of them are errors. 3 instances have been erroneously classified as "intermediate" out of 37 "advanced" type instances. For the 37 "elementary" type, only 4 have been predicted to be "intermediate" and finally, concerning the 37 "intermediate" instances, the system has classified 2 of them as "advanced" and another 2 as "elementary". Under no circumstances has the system predicted an "advanced" instance to be "elementary" or vice versa.

4.4 Discussion: Best Feature Selection and Corpus Analysis

Using both absolute and ratio features results in a higher score (Table 1). However, we have decided to exclude absolute numbers for stylistic analysis of AzterTest, since they may hinder other linguistic and stylistic features. For example, the raw number of words is usually a predictive feature, but it depends on text length and not on its linguistic characteristics. That is, a short text can be simple or complex even though it is short. Measuring features independently of text length allows the user to compare different texts.

Following, in Table 3 we present the the linguistic/stylistic analysis of the corpus, where we show the average values of the 25 most predictive ratios. The abbreviations we use are: n.=number, i.=incidence (per 1000 words), m.=mean, s.=standard deviation, sent.=sentence, prop.=proposition, sub.=subordinate and w.=word.

We observe in this corpus, which compiles journalist texts adapted for ESL readers, that the key features to discriminate among

Feature	A	I	E
Word Frequency			
i. of rare verbs	18.38	11.63	7.57
m. of distinct rare content w.	18.22	13.99	10.73
m. of rare content w.	15.36	12.16	9.76
i. of rare adj.	13.22	10.09	6.51
Descriptive			
s. of w. per sent.	10.66	8.95	7.46
s. of w. per sent. without stop w.	7.47	6.34	5.31
m. of w. per sent.	21.11	18.83	16.14
i. of n. of sent.	48.50	53.90	62.34
m. of w. per sent. without stop w.	14.61	13.00	11.19
s. of letters in w.	2.55	2.49	2.34
Vocabulary			
i. of B2	32.66	27.39	18.42
i. of C1	10.97	7.37	4.30
Lexical Diversity			
Honoré	984.93	896.14	779.31
Maas	0.0506	0.0546	0.0621
MTLD	119.32	106.98	90.09
Syntactic Complexity			
m. of prop. per sent.	52.22	41.65	31.37
m. NP per sent.	6.82	6.18	5.42
m. of punc. per sent.	2.58	2.33	2.04
m. VP per sent.	3.18	2.88	2.55
m. depth per sent.	5.79	5.51	5.11
Syntactic Pattern Density			
i. gerund density	16.42	12.84	7.94
Readability			
Flesch-Kincaid	11.55	10.27	8.59
Flesch Ease	51.54	56.28	63.43
SMOG	8.64	8.04	7.07
Word Semantic			
m. hypernym of verbs	2.09	1.97	1.83

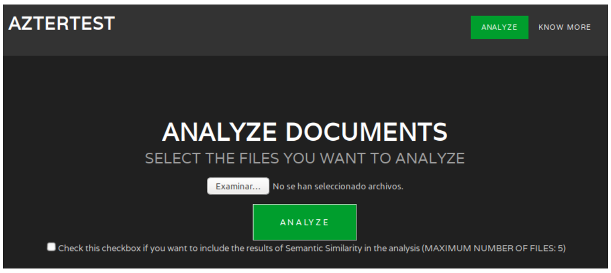
Table 3: Corpus analysis with the 25 best predictive ratios of AzterTest

the three linguistic levels are, a) concerning word frequency, distinct rare content words, particularly verbs and adjectives; b) regarding descriptive features, words per sentence with and without stopwords and letters per word; c) at vocabulary level, incidence of B2 and C1 words; d) and lexical diversity, Honoré, Maas measures and MTLD; e) at syntactic level, propositions, NPs, VPs and punctuation marks per sentence and sentence depth; f) regarding the classical readability formulae, Flesch-Kincaid, Flesch Ease and SMOG; and, finally, g) at semantic the level,the hypermyn verbs index.

All the values decrease from advanced to elementary level, except for the incidence of number of sentences, MAAS lexical density and Flesch Ease. In the case of the MAAS and Flesch Ease, higher scores indicate that the texts are simpler, which correlates to the rest of the features. The higher number of sentences can be explained because simpler texts have shorter sentences and less clauses, and therefore, more sentences are required to communicate the information in the texts.

5 *AzterTest: Web Tool and Source*

Additionally, AzterTest is a web tool that computes 153 features of the linguistic and discourse representations of a text, including descriptive, lexical diversity, readability, word morphological information, word frequency, vocabulary knowledge, syntactic complexity, syntactic pattern density, word semantic information, referential cohesion and connectives. Furthermore, AzterTest web tool classifies the text under three language levels of difficulty (elementary, intermediate and advanced). The tool can be tested in the following website http://178.128.198.190. In Figures 1 and 2 we show the home page of AzterTest and an excerpt of its analysis respectively.



ANALYZE DOCUMENTS

SELECT THE FILES YOU WANT TO ANALYZE

No se han seleccionado archivos.

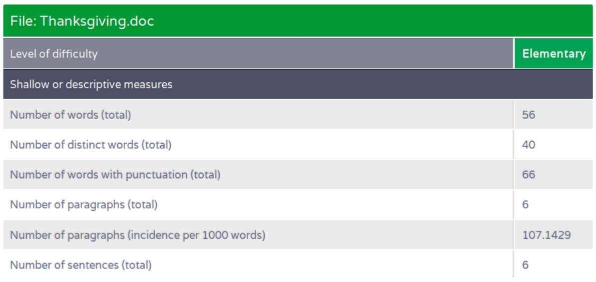
□ Check this checkbox if you want to include the results of Semantic Similarity in the analysis (MAXIMUM NUMBER OF FILES: 5)

Figure 1: Main Page of AzterTest Web Tool

AzterTest source is implemented in Python and it is freely available from a public GitHub repository https://github.com/kepaxabier/AzterTest.

6 *Conclusions and future work*

In this paper, we have introduced AzterTest, an open source linguistic and stylistic analysis tool. AzterTest computes and takes into account 153



File: Thanksgiving.doc

Level of difficulty	Elementary
Shallow or descriptive measures	
Number of words (total)	56
Number of distinct words (total)	40
Number of words with punctuation (total)	66
Number of paragraphs (total)	6
Number of paragraphs (incidence per 1000 words)	107.1429
Number of sentences (total)	6

Figure 2: Screenshot of AzterTest Result (Readability and Descriptive Features)

features, which are grouped into descriptive incidences, word frequencies, vocabulary knowledge, lexical diversity, morphological information, syntactic phenomena, classical readability formulae, semantic information and cohesion devices. The main contributions concerning features are related to new vocabulary and frequency.

Moreover, we have tested AzterTest in a readability assessment scenario for English texts, and using a set of 50 features and the classifier SMO we have obtained an accuracy of 90.09 %. This model outperforms the results obtained with Coh-Metrix's output in this task.

Furthermore, we have made the web application available to teachers so that they can assess the linguistic, stylistic and readability characteristics of their reading materials.

Considering that AzterTest is based on universal dependency parsers, in the future, we will adapt it for multiple languages, and we also plan to extend it with additional vocabulary related features. Additionally, we intend to perform a more extensive assessment of the tool with a group of potential users in order to gather information and adapt AzterTest at their suggestions. Finally, we also plan to use AzterTest for other textual analysis across genres and domains or specialised discourse (Parodi, 2006).

Acknowledgments

We acknowledge following projects: DL4NLP (KK-2019/00045), DeepReading RTI2018-096846-B-C21 (MCIU/AEI/FEDER, UE) and BigKnowledge for Text Mining, BBVA.

References

Aluísio, S., L. Specia, C. Gasperin, and C. Scarton. 2010. Readability assessment for text simplification. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–9. ACL.

Boroș, T., S. D. Dumitrescu, and R. Burtica. 2018. Nlp-cube: End-to-end raw text processing with neural networks. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 171–179.

Cer, D., Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Chall, J. S. and E. Dale. 1995. *Readability Revisited: The New Dale–Chall Readability Formula.* Brookline Books, Cambridge, MA.

Dell'Orletta, F., S. Montemagni, and G. Venturi. 2011. READ-IT: assessing readability of Italian texts with a view to text simplification. In *Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies*, SLPAT '11, pages 73–83. ACL.

Feng, L., M. Jansche, M. Huenerfauth, and N. Elhadad. 2010. A comparison of features for automatic readability assessment. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 276–284. ACL.

Flesch, R. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.

François, T. and C. Fairon. 2012. An AI readability formula for French as a foreign language. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 466–477. ACL.

Gonzalez-Dios, I., M. J. Aranzabe, A. Díaz de Ilarraza, and H. Salaberri. 2014. Simple or complex? assessing the readability of basque texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 334–344, Dublin, Ireland, August. DCU and ACL.

Graesser, A. C., D. S. McNamara, and J. M. Kulikowich. 2011. Coh-Metrix Providing Multilevel Analyses of Text Characteristics. *Educational Researcher*, 40(5):223–234.

Gunning, R. 1968. *The technique of clear writing.* McGraw-Hill New York.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. 2009. The WEKA data mining software: an update.

ACM SIGKDD Explorations Newsletter, 11(1):10–18.

Hancke, J., S. Vajjala, and D. Meurers. 2012. Readability Classification for German using lexical, syntactic, and morphological features. In *COLING 2012: Technical Papers*, page 1063–1080.

Landwehr, N., M. Hall, and E. Frank. 2005. Logistic model trees. 95(1-2):161–205.

Madrazo, I. and M. S. Pera. 2019. Multiattentive recurrent neural network architecture for multilingual readability assessment. *Transactions of the Association for Computational Linguistics*, 7:421–436.

Mc Laughlin, G. H. 1969. SMOG grading-a new readability formula. *Journal of reading*, 12(8):639–646.

Miller, G. A. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

OECD. 2016. *PISA 2015. Results in Focus.* OECD Publishing.

Parodi, G. 2006. Discurso especializado y lengua escrita: Foco y variación. *Estudios filológicos*, (41):165–204.

Petersen, S. E. and M. Ostendorf. 2009. A machine learning approach to reading level assessment. *Computer Speech & Language*, 23(1):89–106.

Platt, J. 1998. Sequential minimal optimization: A fast algorithm for training support vector machines. Technical Report MSR-TR-98-14.

Qi, P., T. Dozat, Y. Zhang, and C. D. Manning. 2019. Universal dependency parsing from scratch. *arXiv preprint arXiv:1901.10457.*

Quispersaravia, A., W. Perez, M. A. S. Cabezudo, and F. Alva-Manchengo. 2016. Coh-Metrix-Esp: A Complexity Analysis Tool for Documents Written in Spanish. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4694–4698.

Scarton, C. and S. M. Aluısio. 2010. Coh-metrix-port: a readability assessment tool for texts in brazilian portuguese. In *Proceedings of the 9th International Conference on Computational Processing of the Portuguese Language, Extended Activities Proceedings, PROPOR*, volume 10. sn.

Si, L. and J. Callan. 2001. A statistical model for scientific readability. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 574–576. ACM.

Speer, R., J. Chin, A. Lin, S. Jewett, and L. Nathan. 2018. Luminosoinsight/wordfreq: v2.2, October.

Vajjala, S. and I. Lucic. 2018. Onestopenglish corpus: A new corpus for automatic readability assessment and text simplification.

Venegas, R. 2008. Interfaz computacional de apoyo al análisis textual:“el manchador de textos”. *RLA. Revista de lingüística teórica y aplicada*, 46(2):53–79.

Štajner, S. and H. Saggion. 2013. Readability Indices for Automatic Evaluation of Text Simplification Systems: A Feasibility Study for Spanish. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 374–382, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Weide, R. 2005. The carnegie mellon pronouncing dictionary [cmudict. 0.6].

Zeman, D. and J. Hajič, editors. 2018. *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies.* ACL, Brussels, Belgium, October.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 69-76 recibido 30-11-2019 revisado 14-01-2020 aceptado 16-01-2020

One stage versus two stages deep learning approaches for the extraction of drug-drug interactions from texts

Comparando enfoques deep learning en una fase y en dos fases para extraer interacciones farmacológicas de texto

Antonio Miranda-Escalada[1], **Isabel Segura-Bedmar**[2]
[1]Barcelona Supercomputing Center, Barcelona, Spain
[2]Universidad Carlos III de Madrid, Leganés, Spain
antonio.miranda@bsc.es, isegura@inf.uc3m.es

Abstract: Drug-drug interactions (DDI) are a cause of adverse drug reactions. They occur when a drug has an impact on the effect of another drug. There is not a complete, up to date database where health care professionals can consult the interactions of any drug because most of the knowledge on DDI is hidden in unstructured text. In last years, deep learning has been succesfully applied to the extraction of DDI from texts, which requires the detection and later classification of DDI. Most of the deep learning systems for DDI extraction developed so far have addressed the detection and classification in one single step. In this study, we compare the performance of one-stage and two-stage architectures for DDI extraction. Our architectures are based on a bidirectional recurrent neural network layer composed of Gated Recurrent Units. The two-stage system obtained a 67.45 % micro-average F1 score on the test set.

Keywords: Relation Extraction, Drug-drug interaction, Recurrent Neural Network, Gated Recurrent Unit

Resumen: Las interacciones farmacológicas (DDI) son una de las causas de reacciones adversas a medicamentos. Ocurren cuando una medicina interfiere en la acción de una segunda. En la actualidad, no existe una base de datos completa y actualizada donde los profesionales de la salud puedan consultar las interacciones de cualquier medicamento porque la mayor parte del conocimiento sobre DDIs está oculto en texto no estructurado. En los últimos años, el aprendizaje profundo se ha aplicado con éxito a la extracción de DDIs de los textos, lo que requiere la detección y posterior clasificación de DDIs. La mayoría de los sistemas de aprendizaje profundo para extracción de DDIs desarrollados hasta ahora han abordado la detección y clasificación en un solo paso. En este estudio, comparamos el rendimiento de las arquitecturas de una y dos etapas para la extracción de DDI. Nuestras arquitecturas se basan en una capa de red neuronal recurrente bidireccional compuesta de Gated Recurrent Units (GRU). El sistema en dos etapas obtuvo un puntaje F1 promedio de 67.45 % en el dataset de evaluación.

Palabras clave: Extracción de relaciones, interacciones farmacológicas, Redes neuronales recurrentes, Gated Recurrent Unit

1 *Introduction*

One of the causes of adverse drug reactions (ADR) is the wrong combination of different drugs. That is, when one drug influences the effect of another. This is known as a drug-drug interaction (DDI). DDIs may have a positive effect on human health. Nevertheless, many DDIs may trigger adverse drug reactions, which can cause health problems and increase healthcare costs.

Despite the efforts in reporting ADRs, such as the monitoring system maintained by the World Health Organization, there is not a single up-to-date database where clinicians can look for all the known DDIs of a drug. Current databases have varying update frequencies, being some of them up to 3 years (Segura-Bedmar, 2010). In addition, most of the information available about DDIs is unstructured, written in natural language in

ISSN 1135-5948. DOI 10.26342/2020-64-8

scientific articles, books and reports.

In this scenario, a challenge has been identified: an automatic system to extract DDIs from biomedical literature is needed to create complete and up-to-date databases with information about DDIs for healthcare professionals. These databases would contribute to reduce adverse drug reactions.

The DDI corpus (Herrero-Zazo et al., 2013) as well as the shared tasks DDIExtraction 2011 (Segura Bedmar, Martinez, and Sánchez Cisneros, 2011) and 2013 (Segura-Bedmar, Martínez, and Herrero-Zazo, 2013) have undoubtedly contributed to the progress of NLP research for the extraction of DDI from texts. Since then, the popularity of this task has rapidly increased over the past few years and a considerable number of papers devoted to the topic is published every year. Early systems (Segura-Bedmar, Martinez, and de Pablo-Sánchez, 2011; Thomas et al., 2013; Chowdhury and Lavelli, 2013; Abacha et al., 2015; Kim et al., 2015) were based on linguistic features combined with classical machine learning algorithms such as SVM (with F-measures around 67%)(Kim et al., 2015). Recently, deep learning methods have triggered a revolution in NLP, demonstrating tremendous success in numerous NLP tasks. The task of DDI extractions has not been oblivious to this revolution. Various deep learning architectures based on Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) have been explored to this task in the last five years, achieving state-of-the-art performance (Dewi, Dong, and Hu, 2017; Sun et al., 2018).

In most of these architectures, the detection and classification of DDIs are carried out in one only stage. The recent deep learning architectures have achieved state-of-art results (around 80% of micro-F1), but with long training times because of their deep architectures. Our hypotehsis is that performing first the detection and then the classification could give better results and reduce the training time. The detection step could significantly decrease the number of instances to classify, and thereby, reducing the training time of the classification task. Our goals of this work are two-fold: (1) to study if a two-stage arquictecture could give better results than an one-stage one, and (2) to test if the use of Gated Recurrent Units (GRUs)(Cho et al., 2014), instead of using Long Short-Term Memory (LSTM) units, can improve the results of DDI extraction. GRUs show better performance on smaller datasets (Chung et al., 2014). Moreover, we also study the effect of different pre-trained word embeddings models (Pyysalo et al., 2013) on the results.

2 State of the Art

This section provides a review of the most recent studies based on deep learning about DDI extraction. Deep CNN models (Dewi, Dong, and Hu, 2017; Sun et al., 2018) have shown the state-of-the-art performance (85% in F1-score).

In addition to CNN, several systems have exploited Recurrent Neural Network (RNN) for the task of DDI extraction. While CNN has proved to be successful in discovering discriminative and meaningful phrases in a text, RNN models are able capable of capturing contextual information and long-term dependencies (Lai et al., 2015), which is very important in our case since the clues about a DDI could appear anywhere in a sentence. Most RNN systems for DDI extraction have used LSTM units, to our knowledge. The standard architecture includes, after preprocessing, an embedding layer, a bidirectional LSTM, a pooling layer and a softmax layer that retrieves a probability for each DDI type. Some systems incorporate a dense layer before the softmax layer, as it also happened in some CNN-based systems.

Zheng et al. (2017) used a RNN and obtained 77.3% F1-score. Input attention mechanism was used only on the word embedding vectors. Also, a part of speech (POS) tag embedding and position embeddings were concatenated to the word. Unlike the most systems for DDI extraction based on deep learning methods, there was not pooling layer between the bidirectional LSTM and the softmax layer.

Despite most works employ LSTM units, the work of Yi et al. (2017) proposed a system utilizing GRU. The system concatenated a word embedding with two position embeddings. Then, embedding vectors were fed into a bidirectional GRU layer. The output of the bi-GRU layer was transformed by an attention-pooling layer, and later by another attention layer. Finally, a softmax layer returned the DDI class probabilities. An F1 score of 72.23% on the DDI corpus was reported.

All these systems have in common that the detection and classification of DDIs are performed in one only step. This is the first work that compares one-stage and two-stages deep learning architectures for DDI extraction.

3 Approach

This section details the corpus employed together with the two approaches tested. The first solution, named one-stage architecture, detects and classifies drug-drug interactions employing one single GRU. On the other hand, the second approach, named two-stage architecture, utilizes two steps: one GRU to detect the DDIs and a second GRU to classify them. In addition, the common preprocessing for both solutions is also described.

3.1 DDI Corpus

The DDI corpus (Herrero-Zazo et al., 2013) is the standard corpus employed in DDI extraction from texts and as such is used within this work. It contains sentences mentioning pharmaceutical substances extracted from texts describing DDIs from DrugBank database (Wishart, 2017) and scientific abstracts from Medline. In total, the DDI corpus contains 33,502 DDI instances and it is highly unbalanced: 85% of instances belong to the negative class -no DDI is present- and 15% of instances belong to the positive class -a DDI is described. DDIs are further divided into 4 categories:

- *mechanism*: it is used for pharmacokinetic mechanisms (e.g. *Grepafloxacin may inhibit the metabolism of theobromine*).
- *effect*: it is used for descriptions of effects (e.g. *In uninfected volunteers, 46% developed rash while receiving SUSTIVA and clarithromycin*) or pharmacodynamic mechanisms (e.g. *Chlorthalidone may potentiate the action of other antihypertensive drugs*).
- *advise*: it contains DDIs written as advise (e.g. *Ocupress should be used with caution in patients (...)*).
- *int*: this type is used when no extra information is provided (e.g. *Interaction of clindamycin and gentamicin in vitro*).

The corpus was randomly divided into two datasets for training and test. In our study, we assume that the drug mentions are already provided. Thus, we focus on the tasks of detection and classification of DDIs from texts.

3.2 Models

Preprocessing is common to both systems proposed. First, instances were transformed to lowercase. Then, punctuation signs were removed. After that, drug blinding was performed. That is, the two drugs involved in the interaction are respectively substituted by “DRUGA” and “DRUGB”, and other drugs mentioned in the sentence are substituted by “DRUGN”.

After these steps, sentences are tokenized (split into separate tokens). As a means of having all sentences of equal length, padding was applied after tokenization. A maximum length of 89 was established, since the longest instance in the training dataset of the DDI corpus contains 89 words.

Finally, position flags are added to every token. They indicate the distance of each token to the two drugs involved in the potential interaction (named “DRUGA” and “DRUGB”). Note that, for the tokens “DRUGA” and “DRUGB”, one of the position flags is zero, since the distance to theirselves is zero.

After preprocessing, every instance is represented as a matrix composed by n tokens. This representation is known as the embedding layer, where each token is represented by a vector that concatenates one word embedding and two position embeddings. We explored the use of two different word embedding models. The first one was a pre-trained word2vec 200-dimensional model, which was trained with biomedical texts taken from PubMed and PubMed Central (PMC). The second pre-trained word embedding model was trained using PubMed, PMC and Wikipedia. So, unlike the first model, it contains information about general knowledge texts. Both embeddings were developed by Pyysalo et al. (2013); the corpus to train the first one contained 5.5B tokens and for the second one 7B tokens. Each position embedding codifies the distance from the word to one of the interacting drugs in the DDI instance. Position embeddings were randomly initialized to 32-dimensional vectors from the uniform distribution U(0,0.01). Then, at this point, every instance, S, is a matrix of vectors, $\vec{g_i}$ such as:

$$\begin{aligned} S &= [\vec{g_1}, ..., \vec{g_N}], \\ \vec{g_i} &= \vec{w_i}||\vec{e1_i}||\vec{e2_i} \end{aligned} \tag{1}$$

3.3 one-stage approach for DDI extraction

First, we describe the first approach for DDI extraction, where the detection and classification tasks are performed in one single step. Once an instance has been preprocessed and represesented by the embedding layer described above, the one-stage system classifies it as either belonging to the *negative* class or to one of the four positive DDI types of the DDI corpus (*effect*, *mechanism*, *advise* or *int*). It employs four layers to classify the instances: embedding, bidirectional GRU, max pooling and output layer (figure 1).

After the embedding layer, a bidirectional recurrent neural network with GRUs ensues. We decided to use GRU with 512 units per direction. The output vectors from both directions, $\overrightarrow{h_{f,i}}$ and $\overleftarrow{h_{b,i}}$, are concatenated:

$$\begin{aligned} \overrightarrow{h_{f,i}} &= \overrightarrow{GRU}(\vec{g_i}), \\ \overleftarrow{h_{b,i}} &= \overleftarrow{GRU}(\vec{g_i}), \\ \vec{h_i} &= \overrightarrow{h_{f,i}}||\overrightarrow{h_{b,i}}, \\ S &= [\vec{h_1}, ..., \vec{h_N}] \end{aligned} \tag{2}$$

Then, the instance tensor, S, is input into a max pooling layer, in which only the timestep of highest magnitude for each feature is kept.

$$\begin{aligned} \vec{q_i} &= max(h_i^i, ..., h_N^i), \\ S &= [\vec{q_1}, ..., \vec{q_N}] \end{aligned} \tag{3}$$

Finally, the instance vector, S, is input into a softmax layer with five output units. It contains 5 neurons employing softmax activation functions. Every neuron returns the probability that the instance belongs to one of the five possible classes, the four DDI types defined in the DDI corpus and the non-DDI type. Therefore, a 5-dimensional output vector represents the confidence the system assigns to each of the five classes. The class with higher confidence is selected as the predicted class.

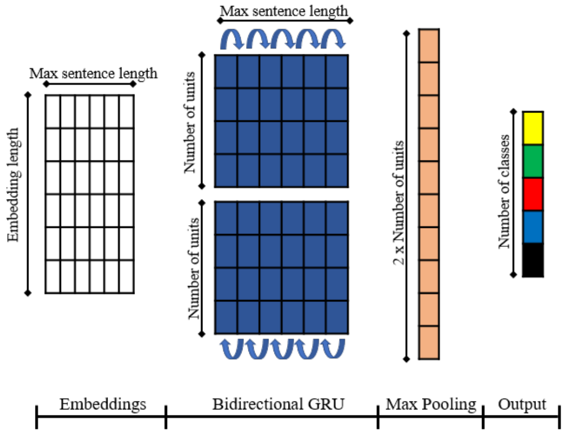


Figure 1: one-stage architecture

3.4 two-stage approach for DDI extraction

Unlike the one-stage system, in the two-stage system, there is a first step to tag each instance as either *negative* or *positive*. Then, instances classified as negative are ruled out. The second step deals with the classification of the positive instances into one of the four DDI corpus categories: *mechanism*, *advise*, *effect* or *int* (figure 2). We describe in more detail each stage below.

The first stage (named detector) also employs four layers to detect the DDI instances: an embedding layer, a bidirectional GRU, a max pooling and a softmax layer with two outputs neurons: one neuron represents the probability for the positive class (the instance is a DDI) and the other neuron stores the probability for the negative class (the instance is not a DDI).

The second stage, named as Classifier, only considers those instances that were classified as positive by the previous step, ruling out the rest of instances. The architecture of this stage is very similar to the previous ones, however, we also introduce, after the max pooling layer, a 64-unit fully connected layer with ReLU activation function, because its integration has shown better performance in combination with complex CNN or RNN architectures (Mohamed, Hinton, and Penn, 2010; Sainath et al., 2015). The output layer contains 4 neurons employing softmax activation functions, one per each DDI type. The higher-probability class is selected as the predicted DDI type.

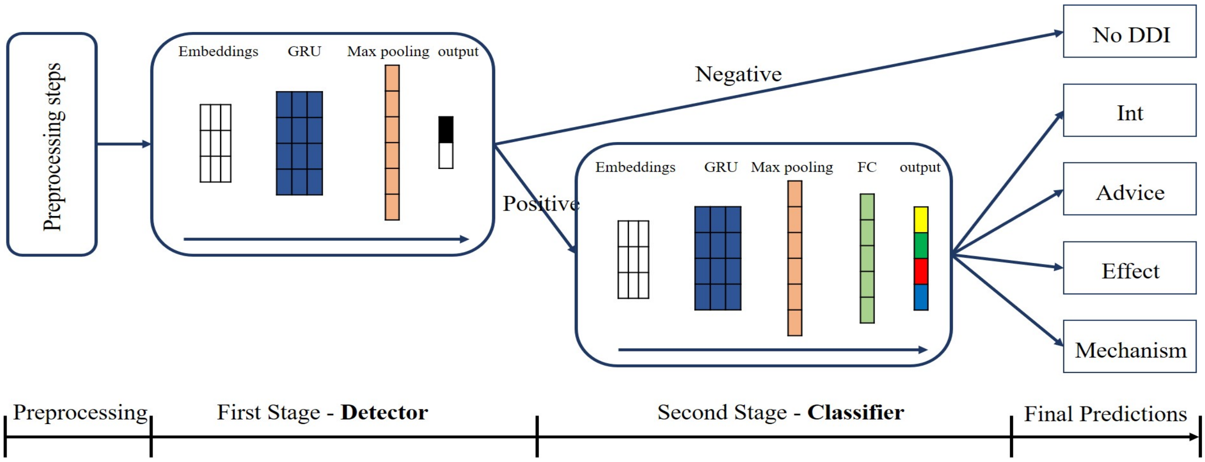


Figure 2: two-stage architecture

3.5 Networks training details

While training the Classifier, class weights were used when computing the loss function for the classes *mechanism*, *advise* and *effect*. This increased the importance on the loss computation of the underrepresented classes and partially solved the class imbalance problem of the dataset. Finally, for training this system, only positive instances were used.

To further deal with class imbalance, negative instance filtering was applied using the rules defined by Kavuluru, Rios, and Tran (2017). In addition, naïve oversampling was employed in the training of the one-stage system and the Detector stage. For the different layers, weights were always initialized according to the Glorot Initializer (Glorot and Bengio, 2010).

In the recurrent layer, we apply always the so-called naïve dropout (Zaremba, Sutskever, and Vinyals, 2015). The dropout probability in the GRU layer and in the fully connected layer was chosen to be always 0.5. The decision was heuristic, since intra-class variability was higher than inter-class variability for the different dropout probabilities tested.

Early Stopping was applied with a patience of 5 epochs. In the Detector stage, F1 score of the positive class was monitored. In the Classifier stage and in the one-stage system, the monitored quantity was micro-average F1 score of the classes *mechanism*, *advise* and *effect*. Class *int* was not taken into account since its number of instances is small (196 in training test, while there are 1687 *effect* instances, for instance).

The optimizer chosen was Adagrad with initial learning rate 0.01. The loss selected was the cross-entropy loss. The mini-batch size was always 50 and the number of epochs was always 25.

4 Results and Discussion

As baseline, we propose the system described in (Suárez-Paniagua and Segura-Bedmar, 2018), which exploited a CNN with max pooling operations, obtaining an F1 of 64.56%.

Table 4 shows the results for both approaches: one-stage versus two-stages architectures. As described above, we aim to compare two different pre-trained word embeddings models to initialize our networks. Both pre-trained models are described in Pyysalo et al. (2013). The first model (from now on, we call it as biomedical) was trained only using biomedical literature such as PubMed and PMC, while the second one (from now on general) was trained also using general texts taken from Wikipedia.

From the results in Table 4, it is seen that the use of GRU is superior to the CNN architecture, since both compared architectures (one-step and two-step) show greater performance than the baseline (Suárez-Paniagua and Segura-Bedmar, 2018), even using general domain word embeddings.

In the one-stage approach, the biomedical pre-trained word embedding models provides slightly better results than using the general pre-trained word embedding model (see Table 4).

Only the precision of the *int* class is better using the general pre-trained word embedding model. However, this class is the least represented in the corpus: there are 189

	One-S. (biomedical)			**One-S. (general)**			**Two-S. (biomedical)**			**Two-S. (general)**		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Int	0.65	0.33	0.44	0.67	0.31	0.43	0.68	0.38	0.48	0.53	0.34	0.45
Advise	0.78	0.75	0.77	0.75	0.71	0.73	0.74	0.79	0.76	0.69	0.80	0.74
Effect	0.68	0.69	0.69	0.66	0.68	0.67	0.64	0.70	0.67	0.61	0.70	0.65
Mechanism	0.72	0.66	0.69	0.72	0.60	0.66	0.68	0.66	0.67	0.66	0.73	0.69
Micro-average	0.71	0.66	**0.69**	0.70	0.62	**0.66**	0.67	0.67	**0.67**	0.64	0.70	**0.67**
Macro-avg.	0.71	0.61	**0.65**	0.70	0.59	**0.62**	0.69	0.63	**0.65**	0.62	0.64	**0.63**

Table 1: Comparative results for one-stage and two-stages systems for DDI extraction

int instances in the training set and 96 in the test set, while there are 1687 and 360 *effect* instances. Therefore, results from of class are less informative of general classifier behaviour.

Finally, the McNemar test (McNemar, 1947) of homogeneity was performed to compare the results of the system using both word embeddings. This test assesses whether the proportion of errors of two classifiers is statistically significant. In this case, it is, since the p-value was smaller than 0.05 (0.02). Therefore, in the case of the one-stage approach, using a word embedding trained with texts from the domain of knowledge of the problem appears to be the optimum choice.

Table 4 also compares the results obtained with the two different word embeddings models for the two-stage approach. The general pre-trained word embedding model obtains a micro-average F1 score of 66.73%, slightly smaller than the metric obtained with the biomedical pre-trained model.

Both word embedding models result in similar performances. However, there are a few differences whose discussion ensue. First, the recall of the *mechanism* class is 7 points higher using the domain-combined word embedding: there are less *mechanism* instances misclassified as other classes. This effect is not observed in the other positive classes. Therefore, there may be a difference in the way information is encoded in *mechanism* sentences and it is affected by the use of a word embedding that incorporates information from domains of knowledge different from the biomedical one.

Second, there are 14 points of difference in the precision for the *Int* class. The reason is a 209% increment in the *negative* instances misclassified as *Int*. However, the total number of *Int* instances is extremely low: the 209% increment corresponds to a change from 11 to 23 misclassified instances.

Finally, the McNemar test of homogeneity has been performed and according to the resulting p-value, p=0.054, the difference in the proportion of errors is on the edge of being significant using the two word embeddings.

We now compare the performance of the one-stage and two-stage approaches. If we use the biomedical pre-trained word embedding model for both approaches, the micro-average F1 score of the one-stage System is 68.54% on the test set, while the two-stage approach achieves an F1 score of 67.45%.

The one-stage approach is superior for all classes (except for *Int*) on the relevant metric, F1 score. Nonetheless, the two-stage system is superior in recall for classes *Int*, *advise* and *effect*. Then, in the one-stage approach, there were more false negatives, while in the two-stage approach, there were more false positives. This phenomenon may be a consequence of the two-stage approach. In it, the first stage (detector) rules out *negative* instances. However, its performance is not perfect, and *negative* instances may be misclassified as *positive* instances, which are entered to the second stage, the classifier. There, these *negative* instances are tagged as *mechanism*, *advise*, *effect* or *Int*. And therefore, the number of false positives increases for the positive classes. An example of this phenomenom is shown in Table 4, sentences 2 and 3.

To statistically compare both systems, McNemar test of homogeneity was performed. The resulting p-value, p=0.147, does not allow to say that both systems are significantly different in their error proportion.

Performances of one-stage and two-stage systems are comparable. The one-stage architecture obtains a slightly better performance. This may indicate that one-stage systems are more suitable for DDI extraction.

Some light experimentation showed that both proposed architectures misclassify sentences with more than two drugs mentioned. As seen in example 1 from Table 4, two drugs adjacent in the sentence are not considered to interact, but when one of them is further away, an interaction is wrongly undetected by

Sentence	**One-S.**	**Two-S.**	**Truth**
(1) **DRUGA DRUGB** as well as other DRUGN may affect ... to DRUGN	None	None	None
DRUGN **DRUGA** as well as other DRUGN may affect ... to **DRUGB**	None	effect	effect
(2) the use of **DRUGA** before **DRUGB** to attenuate ... DRUGN has not been studied	None	effect	None
(3) ... drug drug interaction studies between **DRUGA** and **DRUGB** are inconclusive	None	effect	None
(4) **DRUGA** inhibits the glucuronidation of **DRUGB** and could possibly potentiate DRUGN	None	effect	mechanism
DRUGA inhibits the glucuronidation of DRUGN and could possibly potentiate **DRUGB**	effect	None	effect
DRUGN inhibits the glucuronidation of **DRUGA** and could possibly potentiate **DRUGB**	None	None	None

Table 2: Examples of errors in prediction

the one-stage system. In addition, handling with negation and uncertainty are recurrent challenges in NLP systems. Example 4 from Table 4 shows how both architectures commit mistakes when dealing with the construction *could possible potentiate*. A greater corpus could allow the network to reduce those mistakes and others.

5 Conclusions and Future Directions

Most previous studies on DDI extraction usually opt for one-stage architecture. This work proposes a comparison of one-stage versus two-stage architectures. The two-stage approach first detects the positive instances and rules out the negative instances. Then, only the positive instances are classified in a second stage. Both approaches use GRU, a type of recurrent neural network unit.

Results did not show a significant difference in the error distribution of both systems. However, F1 score is slightly higher for the one-stage System than for the two-stage (68.54% vs 67.45%).

Since performances are comparable, this suggests that the use of one-stage architectures is more suitable in deep learning based DDI extractors because of its simpler design. On the other hand, experiments show that the one-stage architecture requires more training time, since more sentences are synthetically created in the oversampling phase and therefore more instances are used during training. The pre-trained word embedding model created from biomedical literature also provides better performance than the general word embedding model.

As future work, we plan to explore hybrid architectures which exploit advantages of both CNN and LSTM models.

Acknowledgments

This work was supported by the Research Program of the Ministry of Economy and Competitiveness - Government of Spain, (DeepEMR project TIN2017-87548-C2-1-R).

References

Abacha, A. B., M. F. M. Chowdhury, A. Karanasiou, Y. Mrabet, A. Lavelli, and P. Zweigenbaum. 2015. Text mining for pharmacovigilance: Using machine learning for drug name recognition and drug–drug interaction extraction and classification. *Journal of biomedical informatics*, 58:122–132.

Cho, K., B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Chowdhury, M. F. M. and A. Lavelli. 2013. Exploiting the scope of negations and heterogeneous features for relation extraction: A case study for drug-drug interaction extraction. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 765–771.

Chung, J., C. Gulcehre, K. Cho, and Y. Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning.*

Dewi, I. N., S. Dong, and J. Hu. 2017. Drug-drug interaction relation extraction with deep convolutional neural networks. In *IEEE International Conference on Bioinformatics and Biomedicine (BIBM).*

Glorot, X. and Y. Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *13th International Conference on Artificial Intelligence and Statistics (AISTATS).*

Herrero-Zazo, M., I. Segura-Bedmar, P. Martínez, and T. Declerck. 2013. The DDI corpus: An annotated corpus with pharmacological substancesand drug–drug interactions. *Journal of Biomedical Informatics*, 46:914–20.

Kavuluru, R., A. Rios, and T. Tran. 2017. Extracting Drug-Drug interactions with Word and Character-Level Recurrent Neural Networks. In *IEEE International Conference on Healthcare Informatics (ICHI).*

Kim, S., H. Liu, L. Yeganova, and W. J. Wilbur. 2015. Extracting drug–drug interactions from literature using a rich feature-based linear kernel approach. *Journal of biomedical informatics*, 55:23–30.

Lai, S., L. Xu, K. Liu, and J. Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI conference on artificial intelligence.*

McNemar, Q. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12.

Mohamed, A.-R., G. Hinton, and G. Penn. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference.*

Pyysalo, S., F. Ginter, H. Moen, T. Salakoski, and S. Ananiadou. 2013. Distributional Semantics Resources for Biomedical Text Processing. In *Proceedings of LBM 2013.*

Sainath, T. N., O. Vinyals, A. Senior, and H. Sak. 2015. Convolutional, long short-term memory, fully connected deep neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference.*

Segura-Bedmar, I. 2010. *Application of Information Extraction techniques to pharmacological domain: Extracting drug-drug interactions*. Ph.D. thesis, Department of Computer Science, University Carlos III of Madrid.

Segura-Bedmar, I., P. Martinez, and C. de Pablo-Sánchez. 2011. Using a shallow linguistic kernel for drug–drug interaction extraction. *Journal of biomedical informatics*, 44(5):789–804.

Segura-Bedmar, I., P. Martínez, and M. Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Segura Bedmar, I., P. Martinez, and D. Sánchez Cisneros. 2011. The 1st ddiextraction-2011 challenge task: Extraction of drug-drug interactions from biomedical texts. In *Proceedings of the 1st Challenge Task on Drug-Drug Interaction Extraction 2011.*

Suárez-Paniagua, V. and I. Segura-Bedmar. 2018. Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. *BMC bioinformatics*, 19(8):209.

Sun, X., L. Ma, X. Du, J. Feng, and K. Dong. 2018. Deep Convolution Neural Networks for Drug-Drug Interaction Extraction. In *IEEE International Conference on Bioinformatics and Biomedicine (BIBM).*

Thomas, P., M. Neves, T. Rocktäschel, and U. Leser. 2013. Wbi-ddi: drug-drug interaction extraction using majority voting. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 628–635.

Wishart, D. e. a. 2017. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleid Acird Research*, 8.

Yi, Z., S. Li, J. Yu, and Q. Wu. 2017. Drug-drug Interaction Extraction via Recurrent Neural Network with Multiple Attention Layers. In *International Conference on Advanced Data Mining and Applications.*

Zaremba, W., I. Sutskever, and O. Vinyals. 2015. Recurrent Neural Network Regularization. *arXiv preprint arXiv:1409.2329*, 1.

Zheng, W., H. Lin, L. Luo, Z. Zhao, Z. Li, Y. Zhang, Z. Yang, and J. Wang. 2017. An attention-based effective neural model for drug-drug interactions extraction. *BMC Bioinformatics*, 18.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 77-84 recibido 01-12-2019 revisado 17-01-2020 aceptado 19-01-2020

Reflexive pronouns in Spanish Universal Dependencies

Los pronombres reflexivos en las Dependencias Universales en español

Jasper Degraeuwe, Patrick Goethals
Ghent University (Belgium)

Jasper.Degraeuwe@UGent.be
Patrick.Goethals@UGent.be

Abstract: In this paper, we argue that in current Universal Dependencies treebanks, the annotation of Spanish reflexives is an unsolved problem, which clearly affects the accuracy and consistency of current parsers. We evaluate different proposals for fine-tuning the various categories, and discuss remaining open issues. We believe that the solution for these issues could lie in a multi-layered way of annotating the characteristics, combining annotation of the dependency relation and of the so-called token features, rather than in expanding the number of categories on one layer. We apply this proposal to the v2.5 Spanish UD AnCora treebank and provide a categorized conversion table that can be run with a Python script.
Keywords: reflexive pronouns, *se*, Universal Dependencies, AnCora, Spanish

Resumen: En este trabajo, argumentamos que en los actuales treebanks que aplican el formalismo de las Dependencias Universales, la anotación de los reflexivos españoles es un problema sin resolver, que afecta claramente a la precisión y consistencia de los parsers actuales. Evaluamos diferentes propuestas para afinar las diferentes categorías y discutimos los problemas pendientes. Creemos que la solución para estos problemas se puede encontrar en una anotación en múltiples niveles, combinando la anotación de la relación de dependencia y de las características (*features*) de los tokens, en lugar de ampliar el número de categorías en un solo nivel de anotación. Aplicamos la propuesta a la versión española del treebank UD AnCora (v2.5) y proporcionamos una tabla de conversión categorizada que se puede ejecutar mediante un script Python.
Palabras clave: pronombres reflexivos, *se*, Dependencias Universales, AnCora, español

1 Introduction

In recent years, syntactic parsing, the Natural Language Processing (NLP) technique which assigns a syntactic label to words in a sentence, has been integrated in a wide range of NLP applications. Since these applications do no longer require their users to have an extensive technological expertise, the technique has also become widely accessible to language professionals. In fact, parsers such as spaCy or StandfordNLP can be invoked from simple Python scripts, and generate enriched input for developing intelligent text-based applications. Existing NLP tools are usually trained on reference data (treebanks), which are not only growing in number, but also becoming more and more standardized and comparable within and across languages. The Universal Dependencies (UD) project, launched in 2014, plays a crucial role in this context, as it seeks to develop "cross-linguistically consistent treebank annotation for many languages, with the goal of facilitating multilingual parser development, cross-lingual learning, and parsing research from a language typology perspective" (https://universaldependencies.org/introduction, retrieved 24 January 2020; see also Nivre et al. (2016)). UDv2.5 contains 157 treebanks in 90 languages, including previously built treebanks converted into the UD formalism (e.g. the Spanish AnCora treebank, see Taulé, Martí, and

ISSN 1135-5948. DOI 10.26342/2020-64-9

Recasens (2008) and Martínez Alonso and Zeman (2016)).

However, the UD initiative is also a "constantly improving effort" (Martínez Alonso and Zeman, 2016), meaning that annotation guidelines are constantly being fine-tuned over the successive releases of the treebanks (we will work with the latest 2.5 version). Moreover, several annotation issues, which may be problematic from both a cross-linguistic and an intra-linguistic perspective, remain unsolved. For Spanish, one of these issues is the exact annotation of personal pronouns, more in particular of the potentially reflexive pronouns *me*, *te*, *nos*, *os* and *se* (Marković and Zeman, 2018; Silveira, 2016: 115-144). These are very frequent items in Spanish, with *se* occurring, for example, in more than 20% of the sentences in written genres (percentage obtained from corpus research within the Spanish Corpus Annotation Project (Goethals, 2018)), and in almost 30% of the sentences in the test and training sets of Spanish UD AnCora.

The complexity of this issue is also reflected in the output produced by (the current versions of) publicly available parsers, which is often unreliable, inconsistent and/or very coarse-grained. For example, the indirect object in <u>se</u> *lo dije* ('I said this to him/her') is labeled as a passive marker by StanfordNLP; the inherently reflexive <u>*se*</u> *acuerdan de ti* ('they remember you') is direct object in spaCy and again a passive marker in StandfordNLP; or, finally, in the reflexive passive <u>*se*</u> *celebran los cien años del club* ('the 100th anniversary of the club is celebrated') *se* is direct object in both parsers. Incorrect labeling of this kind happens consistently over a wide range of similar constructions with a potentially reflexive pronoun, which implies that it cannot be due to the inherent error rate of the parsers' machine learning algorithm. Rather, faulty annotations in the underlying treebanks are more likely to be at the root of the problem.

Importantly, when trying to solve parser problems, we should not only try to improve annotation consistency and parser accuracy, but also take into account the (cross-)linguistic analyses made in non-computational linguistics (Croft et al., 2017): as parsers become more accessible, more theoretical linguists will use them and evaluate their (linguistic) accuracy and granularity. In this regard, Spanish *se* is a heavily debated subject, with many studies focusing on both the syntactic and the semantic characteristics of the construction (e.g. Mendikoetxea, 1999; Peregrín Otero, 1999; Maldonado, 2008). Although it goes beyond the scope of this paper to discuss all aspects of these analyses, we will briefly come back to this matter in Section 2.3.4.

In the light of the context we have just outlined, with this paper we wish to contribute to a solution for Spanish reflexives by developing an annotation proposal that adheres to the conceptual UD principles (a.o. allowing a satisfactory linguistic analysis, and rapid and consistent human annotation). We will also propose a concrete reannotation of the Spanish UD AnCora treebank, providing exhaustive and categorized conversion tables, and a corresponding Python script to apply these changes to the original treebank files in CoNLL-U format.

Before proceeding to the discussion, it is worth mentioning that we specifically focus on working with UD-based preprocessed corpora in the field of ICALL (Intelligent Computer-Assisted Language Learning), applied to vocabulary learning. Concretely, our purpose is to develop NLP-based corpus query tools that automatically extract authentic usage examples of verbs or nouns, in order to exemplify the constructions in which they are used, and to generate well-targeted vocabulary learning materials.

2 *Reflexives in Spanish Universal Dependencies*

The UD framework provides three key annotation layers by which linguistic constructions can be progressively defined and differentiated: a morphosyntactic Part-of-Speech (POS) tag (limited to a universal set of seventeen tags); a syntactic dependency relation (e.g. subject, direct object, indirect object,…); and a feature set containing additional lexical and grammatical properties (e.g. number or person in the case of pronouns, or tense in the case of verbs).

2.1 Reflexives in current Spanish UD treebanks

The current annotation of reflexives in UD AnCora is as follows:

1. The POS tag is always PRON (which indeed seems the only possible universal tag,

and which we will leave out of the discussion in the remainder of this paper).

2. The feature set includes properties such as “Case” (Acc, Dat), “Person” and “Reflex”, but does not disambiguate “Case”, and only disambiguates “Reflex” in the case of *me, te, nos* and *os,* but not with *se*. As a result, non-coreferential indirect objects such as *el PP no se lo perdona* (‘the PP does not forgive him this’) are still annotated as “Reflex=Yes”. Furthermore, AnCora does not adjust the feature set of the verbal head according to the function of *se* (e.g. by adding the property “Voice=Pass”, see below).

3. Finally, the dependency label is the layer used to actually differentiate between the different uses of reflexive pronouns. Concretely, reflexives can have three values:

- “expl:pass”, used for impersonal constructions such as *en Europa se trabaja mucho* (‘in Europe, people work a lot’) or impersonal passives where there is no subject concordance between the verb and the argument, e.g. *se condena a los culpables* (‘the culprits are convicted’), but not for regular reflexive passives with subject concordance such as *se ve el efecto* (‘the effect is seen’).
- “iobj” (indirect object), used for prototypical coreferential (*Pedro se quita la chaqueta*, ‘Pedro takes his jacket off’) and non-coreferential indirect objects (*no se lo perdono*, ‘I do not forgive him/her this’), but also for some (semi-)lexicalized indirect objects such as *preguntarse si* (‘to ask yourself if’) or *proponerse hacer algo* (‘to intend to do something’).
- “obj” (direct object), used for all cases that are not “expl:pass” or “iobj”, namely regular reflexive passives (see above), prototypical reflexives (*verse a sí mismo*, ‘to see yourself’), and all other (semi-)lexicalized *se* constructions (*materializarse* ‘to become reality’, *morirse* ‘to die’, *moverse* ‘to move yourself’, *...).*

Apart from actual annotation inconsistencies (which are relatively frequent, e.g. ascending up to 30% and 60% of false positives of “expl:pass” and “iobj”, respectively), the main problem with this annotation scheme is its coarse-grained nature. The taxonomy does not allow, for example, distinguishing between passive (*en este volumen se ofrecen textos sobre*, ‘in this volume texts are provided about’) and reflexive uses (*María se ofrece para hacerse cargo del bebé*, ‘María offers herself to take care of the baby’) of the same verb, or between passive (*se incautaron las armas*, ‘the guns were seized’) and inherently reflexive constructions (*la policía se incauta de la armas*, ‘the police seized the guns’). In all these cases, *se* is labeled as “obj”, and the feature sets (both of *se* and of the verbal head) are also equal. For ICALL purposes, this means that the current labels do not enable retrieving targeted examples to illustrate these construction alternations, although they are highly relevant for L2 learners of Spanish.

Interestingly, the multilingual Parallel Universal Dependencies (PUD) treebank for Spanish (created for the 2017 CoNLL shared task and much smaller than AnCora) follows a different strategy: on the one hand, it assigns the same dependency label “compound:prt” to all cases of *se* (which means that all constructions with *se* are conceptualized as a type of multiword expression), but on the other hand, it does introduce a “Voice” feature in the description of the verb and thus manages to distinguish between passives (“Voice=Pass”) and (inherently) reflexives (“Voice=Act”). This solution, however, contrasts with the current UD guidelines for Spanish, which state that “the Voice feature is not used in Spanish because the passive voice is expressed periphrastically” (https://universaldependencies.org/es/index, retrieved 24 January 2020).

2.2 Towards a new annotation of the dependency relations

Recently, Silveira (2016) and Marković and Zeman (2018) formulated several proposals for improving and refining the annotation of reflexives. There seems to be an agreement about the fact that at least the following uses can and should be distinguished:

1. True reflexives, which can be expanded by a focal reflexive *(a/para) sí mismo/a(s),* or could take other non-coreferential objects (e.g. *le*). In these cases, *se* is assigned the dependency label “obj” (*los participantes tienen que inscribirse*, ‘participants have to

register themselves') or "iobj" (*se reservan el derecho a*, 'they reserve for themselves the right to'), depending on the syntactic function.

2. Passive constructions (e.g. *la noticia no se publicó por razones de seguridad* 'the news was not published for safety reasons' or *se recaudan los ingresos fiscales* 'tax revenues are collected'), where there is verbal concordance with the original object of the corresponding non-reflexive transitive verb, and where a transitive process is evoked in which an (unexpressed and perhaps generic) agent acts upon the object. Here, *se* would be annotated as "expl:pass" (note that this does not cover the same constructions as the current annotation).

3. Impersonal constructions, where *se* is combined with an intransitive verb (*en Europa se trabaja mucho*), or with a transitive verb and a nominal that is explicitly marked as accusative (*se condena a los culpables*). Here, *se* would receive the label "expl:impers" (see also Bouma et al. (2018)).

4. Non-coreferential indirect objects where *se* substitutes *le* when it is combined with accusative *lo/a(s)*, as in *se lo pago* ('I pay it to him/her').

In cases 1, 2 and 3, the reflexive use of the construction activates the same event conceptualization as the non-reflexive counterpart, with *se* occupying one of the "obj" roles, and/or blurring the subject role (in the case of passive and impersonal constructions). However, it is obvious that not all reflexives can be classified into one of these categories (in corpus studies the uncontroversial examples would barely account for 50% of the examples).

Therefore, all proposals also include at least a fifth category, namely inherently reflexive verbs, such as *desmayarse* 'to faint', *parecerse a* 'to resemble' or *negarse a* 'to refuse', which are constructions without a clear transitive counterpart. As stated in the UD guidelines, inherently reflexive verbs "cannot exist without the reflexive clitic, and the clitic cannot be substituted by an irreflexive pronoun or a noun phrase. In many cases, an irreflexive counterpart of the verb actually exists but its meaning is different because it denotes a different action performed by the agent". In these cases, *se* receives the label "expl:pv", meaning that *se* is conceptualized as a lexical morpheme (see also the "compound:prt" label in the Spanish PUD treebank).

Clearly, this set of dependency relations offers a far more subtle way of annotating the reflexive forms. However, there remain several issues, which we will discuss in what follows, and which are mainly related to the annotation of the token features of both the reflexive pronoun and the verbal head.

2.3 What about features?

2.3.1 Voice

First, although the current UD guidelines provide that "Voice=Pass" should not be used for Spanish, we are inclined to follow the PUD practice of adding this property to the feature set of the verbal head in the case of the reflexive passive constructions. It seems counterintuitive to mark the reflexive as "expl:pass", without extending this verbal feature to the head of the reflexive. Moreover, as will become clear from the discussion below, the "Voice=Pass" property also enables us to analyze the "umbrella category" of "expl:pv" in greater detail.

2.3.2 Reflexive / reciprocal

Secondly, the UD guidelines do not make a distinction between reflexive and reciprocal readings. The property "PronType=Rcp" does exist, but it is only applied to cases such as German *einander* 'each other', and as a distinctive feature that contrasts with the broad category of personal pronouns ("PronType=Prs"), to which all reflexives belong by definition. Since the reciprocal use of *se* is only one of its many uses, using "PronType=Rcp" would not be an adequate solution for marking this particular use. However, UD does allow personal pronouns to receive an extra feature called "Reflex", but this takes only one possible value, namely "Yes". We would like to propose that, similarly to the annotation of other features such as "Case", "Reflex" accept two possible values, namely "Reflex" and "Rcp". As a result, it would be possible to distinguish between *es importante quererse (a sí mismo)* 'it is important to love yourself' and *es importante quererse (el uno al otro)* 'it is important to love each other', without jeopardizing the unity of the personal pronoun category. As was the case for "Voice=Pass", the "Reflex=Rcp" property will also prove to be useful for analyzing the "expl:pv" cases.

2.3.3 Comitative case

Thirdly, the feature "Case" for reflexive items could be expanded with "Com" (comitative), which is now exclusively used for describing the pronouns *conmigo/contigo/consigo* ('with me, you, him/herself'). Particularly in the case of the verb *llevar* (*llevarse algo [consigo]*, 'to take something with you'), this seems semantically more appropriate than the "Dat" (dative) value, and it can avoid having to identify two "Dat" arguments in examples such as *el Boca* *se* *le llevó un punto al Deportivo* 'Boca took a point from Deportivo with them'.

2.3.4 Features and "expl:pv" constructions

Although the dependency and feature set modifications of sections 2.2-2.3.3 provide a suitable annotation solution for a considerable number of problematic cases, they do not address the annotation of the "expl:pv" category. Clearly, this category covers a wide range of constructions, which, though having a characteristic in common (i.e. that *se* modifies the verbal event structure rather than referring to one of its participants), seem to differ considerably from each other, as is illustrated by the following list:

- *morirse* (adding the nuance of unexpectedness to *morir* 'to die')
- *la gente se manifiesta* ('people are demonstrating')
- *el fenómeno se manifiesta* ('the phenomenon becomes clear')
- *acordarse de algo* ('to remember something')
- *negarse a algo* ('to refuse to do something')
- *se me ocurre que* ('it occurs to me that')
- *ponerse de acuerdo* ('to agree on something')
- *llevarse bien con alguien* ('to get along with someone')

In this regard, it is important to consider a commonly held point of view in Spanish linguistic tradition, namely that reflexives in Spanish activate a so-called "middle voice", in between active and passive voice. One of the most prototypical middle voice contexts are spontaneous processes such as *el problema se manifiesta cada vez más claramente*, which do not carry a truly reflexive (active) meaning, and which exhibit a clear difference with passive constructions, since the agent role has not "faded away" from the profiled event, but is really absent from it. In fact, the middle voice is even considered as the core value of *se*, or, as Maldonado (2008: 155) puts it, "the analysis of the clitic *se* as a reflexive pronoun misrepresents the overall functions that the clitic displays. Instead it is proposed that while having a reduced number of reflexive uses the clitic *se* is a middle voice marker".

One possible solution to capture this middle voice in annotation (a topic which has been left unaddressed in UD guidelines for Spanish) would be to introduce a new dependency relation (e.g. "expl:middle"). However, both Marković and Zeman (2018) and Silveira (2016) take an explicit stance in this matter, pointing out that the distinction between reflexive and passive, on the one hand, and middle voice, on the other, is too subtle and too hard to discern to create a separate "expl:middle" category. Although this may seem a pragmatic rather than a conceptual decision, it should be highlighted that, while in descriptive and theoretical linguistics syntactic categories are often conceptualized as gradual and partially overlapping categories, in the field of NLP tagging and parsing categories are usually of a discrete nature. Therefore, we want to propose an alternative way to handle the diversity of potentially reflexive pronouns (especially of *se*) in this type of construction.

As was already mentioned, the common characteristic of "expl:pv" cases of *se* is that they modify the verbal event rather than referring to one of its participants (or fading away from it, as is the case in "expl:pass"). Starting from the idea that an "expl:pv" modifies an underlying event frame, we believe that an appropriate answer to the problem of accounting for the diversity of the "expl:pv" cases can come from the definition of the features ("Case" and "Reflex" for the reflexive item, and "Voice" for the head). This proposal provides more category distinctions by combining different annotation layers, and not by multiplying the number of tags on one layer.

A good case in point is the verb *manifestar* (Table 1), which has a basic transitive argument structure (*manifestamos nuestros sentimientos*, 'we express our feelings'), and which can be used in a passive frame with an inanimate subject, as in (a) or, more exceptionally, in a true reflexive such as (c). However, there are many examples that would be classified into the category of “expl:pv”, both with inanimate (b) and with animate subjects (d). These two examples are far from being clear-cut passives and reflexives, respectively, and thus would better be labeled as “expl:pv”, but they also clearly differ from each other. Intuitively speaking, the first example seems more passive than the second, and the second more reflexive than the first. We believe that these intuitions can be captured by combining the different annotation layers: (b) and (d) receive the same dependency label “expl:pv”, but their underlying features allow distinguishing between them. Concretely, with transitive verbs “Case” has to be disambiguated between “Acc” and “Dat” (for simplicity we leave “Com” out of the discussion), “Reflex” between “Reflex” and “Rcp”, and “Voice” between “Act” and “Pass”. With *manifestar,* “Case” would be “Acc” in the four cases, since this is the role that the “reflected argument” would play in a non-expletive construction (namely an active transitive construction for (b) or a true reflexive for (d)). “Reflex” would also be “Reflex” in the four cases, but “Voice” would be “Pass” in (b) and “Act” in (d), reflecting the intuition that the core semantic role of the subject is to undergo the process in (b), and to control it in (d).

		Dependency relation	Features reflexive pronoun		Features verbal head
			Case	Reflex	Voice
a	*como se manifestó en el periódico*	expl:pass	Acc	Reflex	Pass
b	*los problemas se manifestaron desde el primer día*	expl:pv			
c	*Dios se manifestó a sí mismo en Cristo*	obj	Acc	Reflex	Act
d	*la gente se manifiesta por tercer día consecutivo; el presidente se manifestó de acuerdo con … (*a sí mismo)*	expl:pv			

Table 1: Feature annotation on passives, reflexives and their corresponding “expl:pv”. Translations: (a) ‘as was said in the newspaper’, (b) ‘the problems became clear from the first day’, (c) ‘God materialized himself in Christ’, (d) ‘people demonstrated for the third consecutive day’; ‘the president said he agreed with the proposal’ (*him/herself)

Crucially, the feature sets link (b) with (a), and (d) with (c), respectively. This means that the expletive reflexive in (b) modifies an inherently passive construction (converting it, prototypically, into a spontaneous process, see the middle voice above), and that in (d), the expletive modifies an inherently reflexive construction, evoking event structures in which it is not relevant to distinguish two separate thematic roles for the reflected argument.

Similarly, the “Case=Acc/Dat” and “Reflex=Reflex/Rcp” properties also enable us to distinguish different underlying structures within the broad category of “expl:pv” examples. As is illustrated in Table 2, the difference between accusative (f) and dative reflexive (h) shows similarities with (e) and (g), respectively, and the difference between accusative reflexive (f) and reciprocal (j) is similar to the difference between (e) and (i).

		Dependency relation	Features reflexive pronoun		Features verbal head
			Case	Reflex	Voice
e	*se ve en el espejo*; *se mete en líos*	obj	Acc	Reflex	Act
f	*se ve amenazado de*; *se mete a hacer algo*	expl: pv			
g	*se quita la ropa*; *se da un baño*	iobj	Dat	Reflex	Act
h	*se da cuenta*	expl:pv			
i	*se saludan*; *se quieren mucho* (*el uno al otro*)	obj	Acc	Rcp	Act
j	*se llevan bien*; *se ponen de acuerdo* (**el uno al otro*)	expl:pv			

Table 2: Feature annotation on accusative/dative reflexives, accusative reflexives/reciprocals and their corresponding “expl:pv”. Translations: (e) ‘he sees himself in the mirror’; ‘he gets himself in trouble’, (f) ‘he is threatened by; ‘he starts doing something’, (g) ‘he takes off his clothes’; ‘he takes a bath’, (h) ‘he realizes something’, (i) ‘they greet each other’; ‘they love each other’, (j) ‘they get along well’; ‘they agree’ (*each other)

2.4 Reannotating Spanish UD AnCora

In Table 3 we present a comprehensive view on the proposed encodings. First, the pronouns were disambiguated according to their general reflexive character, distinguishing between *me veo* (‘I see myself’) and *me ven* (‘they see me’). In the latter group, a distinction is made between “obj” and “iobj” (*me dieron algo*, ‘they gave me something’) at the level of the dependency relation.

Secondly, the reflexive uses were assigned one of the dependency labels “expl:pass”, “obj”, “iobj”, “expl:impers” and “expl:pv”. This means that reflexive and non-reflexive “obj” and “iobj” have the same dependency label but are distinguished by the feature “Reflex”, which is absent in the case of non-reflexives. Reflexive “obj” and “iobj” are further subdivided according to their genuine reflexive versus reciprocal use.

Thirdly, the umbrella category “expl:pv” consists of three subgroups, namely constructions with corresponding transitive verbs, constructions which show an alternation with intransitive verbs, and constructions without corresponding (in)transitive verbs. The first group of “transitivity-based” reflexive constructions is then further subdivided by assigning different combinations of feature sets, as was explained in Section 2.3.4. These feature sets overlap with other “non-expl:pv” constructions, showing their shared characteristics. The proposal also foresees an “expl:pv” category with “Case=Dat”, “Reflex=Rcp” and “Voice=Act”, although this use does not seem to occur in Spanish.

Based on this annotation scheme, we then manually reannotated the AnCora treebank (both the test set and the training set). Table 4 includes a quantitative overview of the original dependency relation labels of all potentially reflexive pronouns (note that “expl:impers” and “expl:pv” do not occur in the original treebank), compared to their new labels after manual reannotation. Apart from the (very numerous) changes in dependency label, it is also worth noting that our reannotation removed the “Reflex” feature from 26 non-coreferential instances of *se*, that adding “Voice=Pass” to the feature set of the verbal head now allows identifying the passive reading of 2715 verb forms, and that, finally, the reciprocal character of 105 pronouns is now reflected in the feature set thanks to the introduction of the “Reflex=Rcp” property.

	Features			
	Pronoun		Verb	
	Case	Reflex	Voice	
Reflexive uses				
expl:pass	Acc	Reflex	Pass	*la noticia se publicó*
obj	Acc	Reflex	Act	*Pedro se ve en el espejo*
	Acc	Rcp	Act	*Pedro y Juan se vieron en la calle*
iobj	Dat	Reflex	Act	*Pedro se quita la ropa*
	Dat	Rcp	Act	*Pedro y Juan se dieron la mano*
expl:impers	-	Reflex	Act	*se trabaja mucho*
expl:pv	***with corresponding non-reflexive transitive verb***			
	Acc	Reflex	Pass	*el fenómeno se manifiesta*
	Acc	Reflex	Act	*la gente se manifiesta*
	Acc	Rcp	Act	*Pedro y Juan se ponen de acuerdo*
	Dat	Reflex	Act	*Pedro se da cuenta*
	Dat	Rcp	Act	*?*
	Com	Reflex	Act	*Pedro se llevó el regalo*
	with corresponding non-reflexive intransitive verb			
	-	Reflex	Act	*Pedro se muere*
	without corresponding non-reflexive verb			
	Acc	Reflex	Act	*Pedro se atreve a …*
Non-reflexive uses				
obj	Acc		-	*me/te/nos/os ven*
iobj	Dat		-	*me/te/nos/os/se lo dijeron*

Table 3: Overview of the annotation scheme for potentially reflexive pronouns in Spanish

reannotated original	expl:impers	expl:pass	expl:pv	iobj	obj	**Total**
expl:pass	285	139	28	1	5	**458**
iobj	1	15	217	142	38	**413**
obj	52	1880	2603	573	618	**5726**
Total	**338**	**2034**	**2848**	**716**	**661**	**6597**

Table 4: Overview of the dependency relation changes in Spanish UD AnCora (test + train)

3 Conclusion

We have argued that in current Spanish Universal Dependencies treebanks, the annotation of reflexives is an unsolved problem. Given the frequency of this construction, occurring for example in more than 20% of the sentences in written texts, this has considerable consequences for parser accuracy and/or granularity. Reflexives, and particularly so-called *se* constructions, have been heavily debated in the tradition of Spanish linguistics. Although it cannot be the aim of morpho-syntactic and dependency parsing to reflect all possible semantic nuances, we have shown that a layered annotation strategy, which combines a relatively limited number of UD dependency relations and feature set properties, can capture both constructional similarities and diversity. We applied this proposal to the v2.5 Spanish UD AnCora treebank and provide categorized conversion tables that can be run as a Python script (see Appendix A and B).

Bibliography

Bouma, G., Hajic, J., Haug, D., Nivre, J., Solberg, P. E., and Øvrelid, L. 2018. Expletives in Universal Dependency Treebanks. In *UDW 2018*, 18-26.

Croft, W., Nordquist, D., Looney, K., and Regan, M. 2017. Linguistic Typology meets Universal Dependencies. In *TLT* 2017: 63-75.

Goethals, P. (2018). Customizing vocabulary learning for advanced learners of Spanish. In T. Read, B. Sedano Cuevas, and S. Montaner-Villalba (Eds.), *Technological innovation for specialized linguistic domains* (pp. 229-240). Berlin: Éditions Universitaires Européennes.

Maldonado, R. 2008. Spanish middle syntax: A usage-based proposal for gramar teaching. In S. De Knop and T. De Rycker (eds.) *Cognitive Approaches to Pedagogical Grammar*, 155-196. Berlin: Mouton De Gruyter.

Marković, S., and Zeman, D. 2018. Reflexives in Universal Dependencies. In TLT 2018.

Martínez Alonso, H. and Zeman D. 2016. Universal Dependencies for the AnCora treebanks. In *Procesamiento de Lenguaje Natural*, 57, 91-98.

Mendikoetxea, A. 1999. Construcciones inacusativas y pasivas. In *Gramática descriptiva de la lengua española*, 2, 1575-1629. Espasa Calpe.

Nivre, J., M.-C. de Marneffe, F. Ginter, Y. Goldberg, J. Hajič, C. Manning, R. McDonald, S. Petrov, S. Pyysalo, N. Silveira, R. Tsarfaty, and D. Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. *LREC* 2016.

Peregrín Otero, C. 1999. Pronombres reflexivos y recíprocos. In *Gramática descriptiva de la lengua española*, 1, 1427-1518. Espasa Calpe.

Silveira, N. 2016. *Designing syntactic representations for NLP: An empirical investigation.* PhD Thesis. Stanford University.

Taulé, M., M. A. Martí, and M. Recasens. 2008. AnCora: Multilevel annotated corpora for Catalan and Spanish. In LREC 2008.

Appendix A: Conversion table

The conversion table includes all occurrences of *me*, *te*, *nos*, *os* and *se*. Other users can modify or customize the annotation decisions.

Appendix B: Python script

The Python script reads in the original CoNLL-U AnCora files, and applies all the changes to the corresponding dependency relations and feature sets. The appended files are available upon request (by email, to Jasper.Degraeuwe@UGent.be).

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 85-92 recibido 01-12-2019 revisado 10-01-2020 aceptado 13-01-2020

A light method for data generation: a combination of Markov Chains and Word Embeddings

Un método ligero de generación de datos: combinación entre Cadenas de Markov y Word Embeddings

Eva Martínez Garcia[1], **Alberto Nogales**[1], **Javier Morales Escudero**[2], **Álvaro J. García-Tejedor**[1]
[1]CEIEC-Universidad Francisco de Vitoria
[2]Avanade Iberia S.L.U.
{eva.martinez, alberto.nogales, a.gtejedor}@ceiec.es, javier.morales.escud@avanade.com

Abstract: Most of the current state-of-the-art Natural Language Processing (NLP) techniques are highly data-dependent. A significant amount of data is required for their training, and in some scenarios data is scarce. We present a hybrid method to generate new sentences for augmenting the training data. Our approach takes advantage of the combination of Markov Chains and word embeddings to produce high-quality data similar to an initial dataset. In contrast to other neural-based generative methods, it does not need a high amount of training data. Results show how our approach can generate useful data for NLP tools. In particular, we validate our approach by building Transformer-based Language Models using data from three different domains in the context of enriching general purpose chatbots.

Keywords: Generation, Hybrid, Markov Chains, Embeddings, Similarity

Resumen: Las técnicas para el Procesamiento del Lenguaje Natural (PLN) que actualmente conforman el estado del arte necesitan una cantidad importante de datos para su entrenamiento que en algunos escenarios puede ser dificil de conseguir. Presentamos un método híbrido para generar frases nuevas que aumenten los datos de entrenamiento, combinando cadenas de Markov y word embeddings para producir datos de alta calidad similares a un conjunto de datos de partida. Proponemos un método ligero que no necesita una gran cantidad de datos. Los resultados muestran cómo nuestro método es capaz de generar datos útiles. En particular, evaluamos los datos generados generando Modelos de Lenguaje basados en el Transformer utilizando datos de tres dominios diferentes en el contexto de enriquecer chatbots de propósito general.

Palabras clave: Generación, Híbrido, Cadena de Markov, Embeddings, Similaridad

1 Introduction

Neural models have become the state-of-the-art for several Natural Language Processing (NLP) approaches such as Machine Translation (MT) (Bahdanau, Cho, and Bengio, 2015; Vaswani et al., 2017; Junczys-Dowmunt, 2019), Dialogue Systems (Sordoni et al., 2015; Vinyals and Le, 2015; Serban et al., 2016; Sankar et al., 2019) or Speech Recognition (Chan et al., 2016; Moritz, Hori, and Roux, 2019; Pham et al., 2019). The most successful ones rely on supervised methods that need a large amount of data. Unfortunately, data are sometimes difficult to obtain, depending on the considered languages or domains. There are several commonly used techniques to perform data augmentation (Tanner and Wong, 1987; Inoue, 2018) like *backtranslation* (Sennrich, Haddow, and Birch, 2016) for MT. We propose a light method to generate extra data to extend a given data set. Our method allows for generating new sentences using Markov Chains (MCs) (Gagniuc, 2017). Then, it filters the generated sentences by using the semantic knowledge enclosed in a word embedding, getting the more adequate ones. We focus our work on the use case of augmenting a corpus used to build a Language Model (LM) that will help to tune chatbots designed for a

ISSN 1135-5948. DOI 10.26342/2020-64-10

specific domain. We validate our approach evaluating the impact of using the generated data to build Transformer-based Language Models by comparing the perplexity [1] of the different models. The experiments show how our MC-based generative method is able to produce adequate sentences since the language models trained using the generated data for a dating domain perform up to 2.71 perplexity points better than the ones trained with only the original data.

The paper is organized as follows. Section 2 briefly explains the state-of-the-art of Natural Language Generation (NLG) and contextualize our approach. Section 3 presents the hybrid MC-word-embedding system revisiting first the main characteristics of each technique. Then, we explain the experiments we carried out to validate our techniques and discuss the obtained results in Section 4. Finally, Section 5 draws conclusions from the presented work and discusses some possible future work lines.

2 Related Work

Natural Language Generation is the task of generating utterances from structured data representations. Data-driven NLG methods facilitate the task of corpus creation since they learn the textual structure and their surface, reducing the amount of human annotation effort. Puzikov and Gurevych (2018) propose a neural encoder-decoder model to participate in the end-to-end *E2E* NLG shared task[2]. Although their neural approach produces fluent utterances, they found out that a template-based model would obtain good results, saving developing and training time. Dušek and Jurčíček (2016) use a *seq2seq*-based generator model in combination with a re-ranking strategy for the n-best output to penalize sentences without required information or that add noise. Further, Liu et al. (2018) introduce a neural approach to generate a description for a table. Their neural model implements a seq2seq architecture consisting of a field-gating encoder, where they update the Long Short Term Memory (LSTM) cell by including a field gating mechanism, and a description generator with dual attention. This dual attention works at word and field level to model the semantic relevance between the generated description and the source table. These neural approaches are effective but they need a high amount of structured data (from 404 to $\sim 700K$ sentences used in the reviewed works). Although there are unsupervised NLG approaches that achieve state-of-the-art results (Freitag and Roy, 2018), they still need a considerable amount of data to train (they use $\sim 256K$ sentences for their unsupervised experiments), preferably in-domain data, which are sometimes scarce.

Similar to the data selection part of our approach, Inaba and Takahashi (2016) present a Neural Utterance Ranking (NUR) model to select candidate utterance according to their suitability regarding a given context. Their model processes word sequences in utterances and utterance sequences in context via Recurrent Neural Networks (RNNs) obtaining good results in ranking utterances more accurately than other methods. They also built a conversational dialog system based on their approach. In contrast, our approach uses a more simple neural model, the *word2vec* (Mikolov et al., 2013) embeddings, to select the more adequate generated sentences since we are not interested in handling the dialog context but in modeling the language that we want a chatbot to produce.

There exist approaches in the area of Dialog Systems that are similar to our method. Wen et al. (2015) use a Stochastic NLG strategy based on a joint RNN and Convolutional Neural Network (CNN). They generate sentences using a forward RNN-LM and then, they use a backward RNN-based LM and a CNN sentence model to re-rank the generated sentences. They can select the most suitable generated utterances without any semantic alignments or predefined grammar trees. Although their approach shows to be effective, they also state the need for a considerable amount of training data, using around 1,300 utterances as training set. In our case, we are dealing with at most hundreds of utterances per domain.

3 Generating More Data

First, we train a Markov Chain from a set of sentences in a given domain. Then, we use it to generate a new set of sentences, replicating the style and using the vocabulary of the

[1] The perplexity is a usual metric to evaluate LMs. It measures how well the language model can predict a word sequence.

[2] `http://www.macs.hw.ac.uk/InteractionLab/E2E`

original data set.

The second step of our approach is to calculate a semantic distance between the generated sentences and the sentences from the original corpus to filter out these sentences that are not sufficiently close to the target domain. Figure 1 depicts the general workflow. We want to discard those new sentences that are semantically too far from the corpus we want to extend since these sentences can add noise to the corpus and may lead to obtaining biased or wrong models in terms of adequacy regarding a given domain.

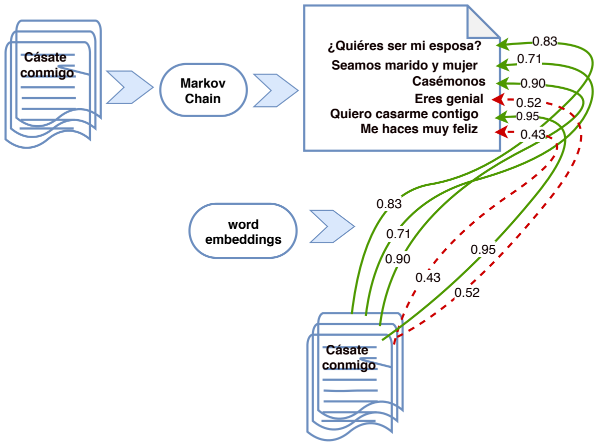


Figure 1: Schema of our generation and filtering method. First, we train a Markov Chain to generate a set of sentences. Then, we use word embeddings to calculate a semantic similarity (values on the arrows) to select the more adequate sentences (not-dashed arrows) and discard the less similar ones (dashed arrows) according to a similarity threshold.

3.1 Sentence generation with Markov Chains

Markov Chains are statistical models well suited for sequence processing. They are useful to compute a probability for a sequence of observable events like words. More formally, a Markov Chain is a probabilistic model that gives information about the probabilities of sequences of random variables or states that can take on values from some set (Jurafsky and Martin, 2008).

These models assume that to predict the future in a sequence all that matters is the current state. Thus, the probability of a state q_t taking on the value a can be expressed as follows[3]:

$$P(q_t = a|q_1 \dots q_{t-1}) = P(q_t = a|q_{t-1})$$

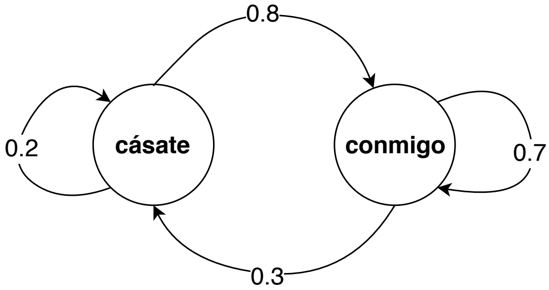


Figure 2: Two-state Markov Chain diagram. The states take values on a vocabulary in the *dating* domain. Each edge expresses the probability of generating a particular word given the preceding one.

For instance, Figure 2 shows how a two-state Markov Chain can model a proposition sentence like *"cásate conmigo"*, which is *"marry me"* in Spanish, as a sequence of words. Note that the edges represent the probabilities of generating *"cásate"* or *"conmigo"* depending on the word generated first.

In our approach, we take advantage of these properties of the Markov Chains to learn the style particularities from a given data set. Then, we generate a new set of sentences using the word probability distribution learned by the MC from the original in-domain corpus.

3.2 Data Filtering with word embeddings

The generated sentences that are more similar to a target domain corpus will help us to obtain more adequate NLP tools or models given a specific domain.

Word embeddings (Mikolov et al., 2013; Peters et al., 2018; Devlin et al., 2019) are distributed word representation models, typically based on neural networks, that are able to capture words' semantic information. These models have proved to be robust and powerful for predicting semantic relations between words and even across languages (Artetxe, Labaka, and Agirre, 2017; Devlin et al., 2019; Ruiter, España-Bonet, and van Genabith, 2019)

[3] This also represents a bigram language model, where the conditional probability of the next word is approximated by using only the conditional probability of the preceding word.

Following a usual approach to work with word embeddings and semantic distance, given a sentence $s = w_1 w_2 \dots w_t$, we define its vector representation as the resulting vector from the average of the vectors for each word in the sentence:

$$\vec{s} = \frac{1}{t}\sum_{1}^{t} \vec{w_i}$$

Then, we use the cosine similarity to measure the semantic relatedness between the generated sentences s_{gen_i} and the sentences in the target domain corpus $S_{tdomain}$:

$$cossim(\vec{s_{gen_i}}, \vec{S_{tdomain}}) = \frac{\vec{s_{gen_i}} \cdot \vec{S_{tdomain}}}{\|\vec{s_{gen_i}}\| \|\vec{S_{tdomain}}\|},$$

being $\vec{S_{tdomain}}$ the vector representation of the target domain data calculated as the average of the vector representation for each sentence in S_{orig}, which are calculated, as before, as the average of the vectors for each word in the sentence. Note that the cosine similarity takes values in $[-1, 1]$, understanding that a higher value will indicate a higher semantic closeness. We discard every s_{gen_i} that is not close enough to the corpus in the target domain. In other words, we only keep those s_{gen_i} that their *cossim* with the vector of the target domain corpus $S_{tdomain}$ is greater than a fixed threshold. This threshold will be set experimentally for each processed corpus.

4 *In-Domain Language Models*

In order to validate the data generated by applying our approach, we build several LMs: a baseline on a usual subtitles corpus, one on each in-domain original corpus and one on each generated in-domain corpus. Then, we evaluate them by calculating their perplexity on their corresponding in-domain test set.

4.1 Data Generation Settings

We generate new data for three different domains. Each original domain corpus contains a set of utterances[4] gathered from a real running chatbot and reflects different users' interactions. The *dating* domain corpus has sentences that can appear in romantic conversations. The *recipes* domain corpus gathers sentences that express user's recipes preferences. And finally, the *livefb* corpus contains sentences with living queries and mentions to Facebook[5]. We kept 50 sentences from each domain corpora as test set for the LM evaluation we will pursue later on.

threshold	dating	recipes	livefb
original	131	94	89
0.50	78	315	156
0.60	78	315	**156**
0.70	78	315	154
0.75	**78**	**315**	154
0.8	77	310	154
0.85	72	308	154
0.9	69	293	144
0.95	38	250	104
0.98	1	100	26
concat	209	409	245
vocab	192	137	135

Table 1: Number of unique generated sentences for different domains using different thresholds. The *original* row is for the number of unique sentences for each original in-domain corpus. The *concat* is for the number of unique sentences after concatenating the selected generated dataset (in bold) plus the original in-domain corpus. The *vocab* row is for the vocabulary size for each domain data.

We train a two-states MC to generate new sentences with Markovify[6] for each domain. Table 1 shows the number of unique sentences used as MC training set for each domain in the *original* row. In particular, we generate up to 1,000 sentences in our experiments using the MC to filter out afterward the more adequate ones.

For the filtering task, we use the *es_core_news_md*[7] word embedding model available in the *spacy* library[8]. It is a multitask CNN-based word embedding trained on the AnCora (Taulé, Martí, and Recasens, 2008) [9] and WikiNER (Ghaddar and Langlais, 2017) corpora. We carry out a grid

[4] We understand an utterance as a dialog act, in the context of a conversational dialog, that serves a function in the dialog.

[5] www.facebook.com

[6] https://github.com/jsvine/markovify

[7] https://spacy.io/models/es#es_core_news_md

[8] https://spacy.io/

[9] http://clic.ub.edu/corpus/ancora

domain	**original sentence**	**generated sentence**
dating	¿Quieres ser mi esposa?	¿Te gustaría ser mi esposa?
recipes	Confío en ti, ¿Me recomiendas la mejor receta?	Confío en ti, ¿Me ayudas a elegir una receta?
livefb	Me dijeron que estás en China, ¿es cierto?	Me dijeron que estás en China, ¿En qué lugar vives?

Table 2: Examples of generated sentences for each of the studied domains in comparison with sentences from the original datasets.

search to adjust the similarity threshold value to keep the more adequate sentences.

Table 1 shows the figures of the different generated data. As long as the similarity threshold increases, the number of filtered sentences decreases as expected. For thresholds below 0.75, the number of generated sentences is the same as per 0.75. For *dating* and *recipes* domains, our method generates the same number of sentences for all thresholds bellow 0.75 whereas for the *livefb* domain it is for threshold 0.60. In particular, we choose these values for the similarity thresholds respectively for each domain, using the resulting generated sentences to build the training corpora by concatenating them to the original in-domain sentences. On the other hand, the number of unique generated sentences by the MC for each domain coincides with the number of sentences indicated in Table 1 for the lower values of the similarity threshold. These facts indicate that the Markov Chain generates sentences that are semantically related to the original domain. This is expected since Markov Chains were built using only these data, sharing then the vocabulary.

Furthermore, it is noticeable that there is only a small overlapping between the original corpus and the generated sentences as shown in the *concat* row in Table 1, that shows the number of unique sentences after concatenating the selected generated dataset plus the original in-domain corpus. These numbers reflect that the generation method is able to propose new adequate sentences. Table 2 shows some examples of generated examples for the different domains.

4.2 Language Models Settings

All the LMs that we built are Transformer-based Language Models trained using Marian (Junczys-Dowmunt et al., 2018) with 128 dimensional embeddings and hidden layers with 256 units. We build a baseline LM using the Spanish side of the English-Spanish OpenSubtitles2018 corpus (Tiedemann, 2012) as training set (61,434,251 sentences), fixing a vocabulary size of 50,000. We kept the last 1,000 sentences as out-domain test set. We chose this corpus to build a reference baseline since it is a collection of movie subtitles and thus they are close to the language particularities of the utterances we want to handle.

4.3 Evaluating the Language Models

We carried out a simple evaluation task. We obtained the perplexities [10] of the different models on their corresponding in-domain test sets of 50 dating utterances each, shown in Table 3.

model	**dating**	**recipes**	**livefb**
baseline	28,37	92.99	52.54
original	28.69	**5.28**	**5.55**
original++	**25.98**	5.76	5.68

Table 3: Perplexity values on the in-domain test sets (the lower the better). The *baseline* row is for the LM trained on OpenSubs2018, the original row is for the LM trained on the original in-domain training dataset and the original++ is for the LM trained using the in-domain corpus including the newly generated sentences using the selected threshold for each domain.

It is easy to observe the importance of having in-domain data. Recall that the baseline LM was trained on millions of sentences from subtitles whereas the in-domain LMs were trained using only hundreds of sentences. The LM trained only using the in-domain corpus achieves almost the same perplexity values as the baseline LM for the *dating* domain. Whereas for the other two domains, the LMs trained using only the original in-domain data highly improve the perplexity

[10] The more information an LM gives about a word sequence the lower the perplexity. Better LMs can help to select a more adequate answer in a chatbot workflow.

values. A possible reason for that could be the specificity of the corpora for these two domains in comparison with the data in the *dating* domain. More open domains have larger vocabularies and a higher variability margin that results in obtaining LMs with lower perplexities. In our case, *livefb* and *recipes* domains have fewer data and smaller vocabularies. Thus, it is easier to obtain lower perplexities in these domains than in larger domains like for the *dating* case.

The model trained on the *dating* extended corpora achieves better perplexities than the LM on the *dating* corpus, also better than the baseline LM. For the *recipes* and *livefb* domains, the LMs trained on the extended corpora achieve a similar perplexity than the ones for the LMs trained on the original in-domain data. Note that the baseline LM also achieves the worst perplexities on the *recipes* and the *livefb* test set. These results support also the fact that the *dating* domain data represents a more open domain than the *recipes* and *livefb* ones. Thus, being easier to improve the results achieved using only the original *dating* dataset than in the other two scenarios. Therefore, the results clearly show the importance of the adequacy of the data regarding a specific domain. Furthermore, the numbers also indicate the impact of the specificity of a domain, being more necessary to generate data for more open domains than for the more specific ones.

The best LM in the *dating* domain, the more open one, is the model trained on all the generated sentences, getting a 2.71 points better perplexity. This shows the usefulness of the sentences generated by our method even though having a small original in-domain corpus as a starting point.

5 Conclusions

We propose a light hybrid method to generate extra data to extend a corpus for a specific domain. Our approach is simple yet effective, and it does not need a large amount of data. Our method comprises two phases: first, it uses a Markov Chain to generate sentences. Then, it filters the most similar sentences according to the cosine similarity of their vector representation. The generation method is able to create a significant amount of new sentences with a small overlapping with the original in-domain corpus.

We assess the validity of the generated data by evaluating a set of in-domain LMs trained using a corpus extended with the data generated by applying our method. We found out that our method works well when dealing with data from more open domains. The LMs trained for the *dating* domain, using the data generated by our approach, show the highest quality gain in terms of perplexity. In contrast, the impact of the generated data for LM models on more specific domains, like the *recipes* and *livefb* ones, is not as noticeable since it is more difficult to achieve lower perplexities in this kind of scenario because they are more predictable.

As future work, we want to make a better evaluation of our method using more data, both for training and testing, as soon as they are available. Performing also an external evaluation of the LMs trained using the data generated by our method by including them in a reranking procedure for generating the answer of a chatbot.

We are interested in exploring variations of our method that can lead to quality improvements. Improve sentence representations by using sentence embeddings (Reimers and Gurevych, 2019), (Le and Mikolov, 2014). Generate better utterance candidates by using trigram or 4-gram CMS or even using neural-based generative approaches (Puzikov and Gurevych, 2018; Liu et al., 2018; Bahdanau, Cho, and Bengio, 2015). Also, we want to refine our sentence filtering approach by using other similarity measures like some margin-based scores (Artetxe and Schwenk, 2019) or the CSLS (cross-domain similarity local scaling) (Lample et al., 2018).

Furthermore, we would like to study the impact of using sentences generated using our method to fine-tune the newest word representation models, like BERT (Devlin et al., 2019), ELMO (Peters et al., 2018) or XLNET (Yang et al., 2019), for the language modeling task.

Acknowledgments

We would like to thank Francisco del Valle Bas, Ángel Melchor and Miguel Pajares for their assistance during the development of this research work.

References

Artetxe, M., G. Labaka, and E. Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the ACL2017 (Volume 1: Long Papers)*, pages 451–462.

Artetxe, M. and H. Schwenk. 2019. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the ACL2019 – Volume 1*, pages 3197–3203.

Bahdanau, D., K. Cho, and Y. Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of ICLR 2015*.

Chan, W., N. Jaitly, Q. Le, and O. Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Proceedings of IEEE ICASSP 2016*, pages 4960–4964.

Devlin, J., M. Chang, K. Lee, and K. Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the NAACL-HLT 2019*, pages 4171–4186.

Dušek, O. and F. Jurčíček. 2016. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. In *Proceedings of the ACL2016 (Volume 2: Short Papers)*, pages 45–51.

Freitag, M. and S. Roy. 2018. Unsupervised natural language generation with denoising autoencoders. In *Proceedings of the EMNLP 2018*, pages 3922–3929, October-November.

Gagniuc, P. A. 2017. *Markov chains: from theory to implementation and experimentation*. John Wiley & Sons.

Ghaddar, A. and P. Langlais. 2017. WiNER: A Wikipedia annotated corpus for named entity recognition. In *Proceedings of the IJCNLP 2017(Volume 1: Long Papers)*, pages 413–422.

Inaba, M. and K. Takahashi. 2016. Neural utterance ranking model for conversational dialogue systems. In *Proceedings of the SIGDIAL 2016*, pages 393–403.

Inoue, H. 2018. Data augmentation by pairing samples for images classification. *arXiv preprint arXiv:1801.02929*.

Junczys-Dowmunt, M. 2019. Microsoft translator at wmt 2019: Towards large-scale document-level neural machine translation. In *Proceedings of the WMT19 Translation (Volume 2: Shared Task Papers, Day 1)*, pages 225–233.

Junczys-Dowmunt, M., R. Grundkiewicz, T. Dwojak, H. Hoang, K. Heafield, T. Neckermann, F. Seide, U. Germann, A. Fikri Aji, N. Bogoychev, A. F. T. Martins, and A. Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121.

Jurafsky, D. and J. H. Martin. 2008. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Nueva Jersey: Prentice Hall.

Lample, G., A. Conneau, M. Ranzato, L. Denoyer, and H. Jégou. 2018. Word translation without parallel data. In *Proceedings of the ICLR 2018*.

Le, Q. and T. Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196.

Liu, T., K. Wang, L. Sha, B. Chang, and Z. Sui. 2018. Table-to-text generation by structure-aware seq2seq learning. In *32nd AAAI Conference on Artificial Intelligence*.

Mikolov, T., K. Chen, G. S. Corrado, and J. Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Moritz, N., T. Hori, and J. L. Roux. 2019. Unidirectional Neural Network Architectures for End-to-End Automatic Speech Recognition. In *Proc. Interspeech 2019*, pages 76–80.

Peters, M. E., M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings NAACL-HLT 2018 – Volume 1*, pages 2227–2237.

Pham, N.-Q., T.-S. Nguyen, J. Niehues, M. Müller, and A. Waibel. 2019. Very Deep Self-Attention Networks for End-to-End Speech Recognition. In *Proc. Interspeech 2019*, pages 66–70.

Puzikov, Y. and I. Gurevych. 2018. E2E NLG challenge: Neural models vs. templates. In *Proceedings of the INLG 2018*, pages 463–471.

Reimers, N. and I. Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the EMNLP-IJCNLP 2019*, pages 3982–3992.

Ruiter, D., C. España-Bonet, and J. van Genabith. 2019. Self-Supervised Neural Machine Translation. In *Proceedings of the ACL 2019, Volume 2: Short Papers.*, pages 1828–1834.

Sankar, C., S. Subramanian, C. Pal, S. Chandar, and Y. Bengio. 2019. Do neural dialog systems use the conversation history effectively? an empirical study. In *Proceedings of the ACL2019*, pages 32–37.

Sennrich, R., B. Haddow, and A. Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the ACL2016 (Volume 1: Long Papers)*, pages 86–96.

Serban, I. V., A. Sordoni, Y. Bengio, A. Courville, and J. Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *30th AAAI Conference on Artificial Intelligence.*

Sordoni, A., M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, and B. Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the NACCL-HLT 2015*, pages 196–205.

Tanner, M. A. and W. H. Wong. 1987. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540.

Taulé, M., M. A. Martí, and M. Recasens. 2008. AnCora: Multilevel annotated corpora for Catalan and Spanish. In *Proceedings of the LREC'08.*

Tiedemann, J. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the LREC2012*, pages 2214–2218.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. 2017. Attention is all you need. In *Proceedings of the NIPS2017*, pages 6000–6010.

Vinyals, O. and Q. V. Le. 2015. A neural conversational model. In *Proceedings of the 31 st International Conference on Machine Learning*, volume 37.

Wen, T.-H., M. Gašić, D. Kim, N. Mrkšić, P.-H. Su, D. Vandyke, and S. Young. 2015. Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking. In *Proceedings of the SIGDIAL2015*, pages 275–284.

Yang, Z., Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *ArXiv*, abs/1906.08237.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 93-100 recibido 01-12-2019 revisado 14-01-2020 aceptado 17-01-2020

Can *translationese* features help users select an MT system for post-editing?

¿Pueden ayudar las características del traduccionés *a los usuarios a seleccionar un sistema de TA para posedición?*

Nora Aranberri
IXA Group - University of the Basque Country
nora.aranberri@ehu.eus

Abstract: This work explores the possibility of using translationese features as indicators of machine translation quality for users to select an MT system for post-editing assuming that a lower level of translationese will reveal a reduced need for editing. Results reveal that translationese and automatic metrics rank systems differently, opening an avenue for further research into the information each provides.

Keywords: Machine translation, translationese, quality evaluation

Resumen: Este trabajo explora la posibilidad de utilizar las características del traduccionés como indicadores de calidad de traducción automática para ayudar a los usuarios a seleccionar un sistema de TA para posedición asumiendo que un nivel más bajo de traduccionés revela una menor necesidad de edición. Los resultados apuntan a que el traduccionés y las métricas automáticas clasifican los sistemas de manera diferente, abriendo nuevas vías de investigación sobre la información que aporta cada métrica.

Palabras clave: Traducción automática, traduccionés, evaluación de la calidad

1 Introduction

Translation research has long debated the existence of *translationese*, that is, a set of features common to translated texts that differentiates them from texts originally written in their respective language. Whereas the specific features in which translations and original texts differ is still debated, authors seem to agree that differences exist (Laviosa, 2002), and what is more, that original and translated texts can be automatically distinguished (Baroni and Bernardini, 2005; Volansky, Ordan, and Wintner, 2013). Research suggests that translated texts show signs of reduced richness and seem *'abnormally normal'* (Tirkkonen-Condit, 2002, p. 217). When post-editing machine translation, these translationese features seem to get amplified (Toral, 2019) probably because post-editors are primed by the MT output (Green, Heer, and Manning, 2013), which inherits a significant closeness to the source text and covers as much of the target language the training resources and techniques allow.

While translationese does not intend to refer to poor translated versions that result out of lack of translation skills but rather to features that are intrinsically present in translations and that might even be unavoidable (Tirkkonen-Condit, 2002), when exacerbated, as it seems to be the case when machine translation is involved, the differences between the translation and the original language could indicate a variation in translation quality. To be more precise, we wonder whether translation universals, namely, simplification, normalisation, explicitation and interference (Baker, 1993; Toury, 2012) could provide machine translation users with information about output quality.

So far, in research and professional environments, information about the overall quality has been reported via automatic metrics, and when feasible given the time and cost involved, human evaluations. If proven useful, translationese features would offer an advantage over automatic metrics in that they do not require a reference translation of the segment to be assessed, but rather rely on the difference between the translation of such segment, and standardised features and ratios of the source and target languages.

Recent work on exploring translationese features in post-edited (PE) and human translations has shown evidence that (1) PE

ISSN 1135-5948. DOI 10.26342/2020-64-11

texts are simpler in terms of lexical use, (2) sentence length is closer to the source text length, and (3) grammatical units in PE texts tend to preserve typical sequences of the source language to a larger extent that in human translations (Toral, 2019). This seems to indicate that, as Green, Heer, and Manning (2013) claim, translators are primed by the MT output. This being so, there are grounds to believe that an MT output with a lower level of translationese might benefit users who intend to edit the MT output.

In this work, we aim to analyse whether differences in translationese are present in the output of different MT systems and whether this could be used to help a user select the system that suits a particular text better. To that end, we set the experiment within the current scenario in the Basque Country, where multiple MT systems have been made available to the public in a relative short period of time. We collect the output of the systems and study the relation between translationese features and automatic metrics. Results suggest that they do not always point at the same MT system as the best performing. This outcome opens up a new avenue for further investigation about the type and usefulness of the information provided by translationese features, specially for selecting a system for post-editing.

2 The Basque context

According to a market study conducted by Langune, the Basque Association of Language Industries, in the Basque Autonomous Region alone, the translation industry market reached 40 million euros in 2017 with the translation of around 500 million words, when considering the demand of both the public and private sectors [1]. Needless to say that this study did not consider the additional translation needs that are addressed without resorting to professional services. Translations between the two official languages of the region, namely, Basque and Spanish, are commonplace in both work environments and the private sphere.

Publicly available systems are emerging. In less than a year, three different systems have joined Google Translate[2] in offering Spanish-Basque neural machine translations, namely, Modela[3], Batua[4], and the neural system offered by the Basque Government[5][6].

The systems are made available under the assumption that they provide translation quality useful for users. These users might be professional translators that are willing to try the system to see whether a paid subscription or a customised version would be worth investing in, or regular users who find themselves having to translate texts in their daily lives or even professional settings, and given the free nature of the systems, decide to try them out. All systems include warning messages about the potential quality issues of the output and specify that it should be used carefully. However, they do not provide any further indication as to the quality the system offers. Indeed, this is very difficult to evaluate, as a system's performance might vary considerably depending on the text used as input, its syntactic complexity, its topic, its register, that is, how suitable the text is for the particular system to translate given the information used during its training.

From a user's perspective, what is the difference between the systems in terms of quality? Should each user pick a system after attempts of trial and error? Is it possible to provide users with some pointers as to which one to select depending on their needs? This work is a first step towards studying the output of the available systems.

3 Experimental set-up

The following subsections describe the data sets used and the MT systems tested in this work.

3.1 MT systems

The MT systems used in this study are five freely available systems, four neural systems and a rule-based system, that can translate from Spanish into Basque. It is worth noting that we are not aware of the data used to train the different systems and that it is possible that part of the data sets used in the experiments were included during training. However, we believe that this does not pose a threat to this study because it is not

[1] http://www.langune.eus

[2] Available at: https://translate.google.com

[3] Available at: https://www.modela.eus/eu/itzultzailea

[4] Available at: https://www.batua.eus/

[5] Available at: http://www.euskadi.eus/itzultzailea/

[6] Note that following the completion of this research, in December 2019, the freely available NMT system *itzultzailea* was launched by Elhuyar.

our objective to discover the systems' absolute translation quality but rather compare the results of translationese features and automatic metrics.

- **Google Translator.** This is the multilingual multidirectional MT system service developed by Google. It is not clear from the documentation available if the queries for Spanish–to–Basque translations are handled by a "zero-shot" neural system, by linking two neural systems, that is, Spanish-English, English-Basque, or with phrase-based statistical systems. The number of words to be translated freely through the web is unlimited but a user may not translate more than 5,000 words at a time, which can be input into the system window or uploaded as a document.
- **Modela.** It is a bidirectional Spanish-Basque neural MT system developed in a research project funded by the Elkartek scheme of the Basque Government and the Basque Business Development Agency during 2016-2017 (Etchegoyhen et al., 2018). The consortium consisted of ISEA (Coordinator of innovation projects for MondragonLingua), the IXA research group of the University of the Basque Country, the Vicomtech research centre, the Ametzaigaina technological agent and the Elhuyar Foundation. In November 2018, the consortium agreed to have a baseline system publicly available and users may translate up to 2,000 words per month free of charge.
- **Batua.** It is a bidirectinal Spanish-Basque neural MT system powered by Vicomtech and sponsored by the telecommunications group Euskaltel and the linguistic services group MondragonLingua. It was released in 2019. The number of words to be translated freely is unlimited but a user may not translate more than 1,000 words at a time, which must be input into the system window. It is stated that the system is in beta version. It provides the user with the possibility to edit and correct the translations, which the system will use to improve the quality of its output.
- **EJ-NMT.** It is a bidirectional Spanish-Basque neural MT system made available by the Department of Culture and Language Policy of the Basque Government, *Eusko Jaurlaritza*, *EJ* for short. The system was developed in collaboration with departments within the EJ, the Basque Network for Science and Technology and other organisations with a strong focus on language, such as the Basque radio and television broadcaster EITB. Additionally, the system has been customised with the translation memories created over the last 20 years at the Basque Institute for Public Administration. It was released on October 16, 2019. The number of words to be translated freely is unlimited but a user may not translate more than 4,000 characters at a time. The text can be typed into the system window or a URL may be provided. The page displays a warning stating that the system is in beta version.
- **EJ-RBMT.** It is a rule-based MT system made available by the Department of Culture and Language Policy of the Basque Government, *EJ*. The system has been running since 2010 powered by Lucy and supports translation between Spanish and Basque, and English and Basque. The number of words to be translated freely is unlimited but a user may not translate more than 1,500 characters at a time. The text can be typed into the system window or a URL may be provided. The page includes the option of displaying multiple translations for ambiguous words.

The use of the five systems will allow us to study whether differences between the systems in terms of translationese are considerable, and also analyse the behaviour of translationese-related metrics in NMT and RBMT systems. Translations for all systems and all data sets were collected over the first two weeks of November 2019.

3.2 Data sets

The data sets to be used must comply with a number of criteria to be adequate for the study. Firstly, we require data sets from different domains in order to check whether systems perform differently depending on the topic. We aimed at using existing publicly available sets whenever possible to facilitate replication. However, due to the limited sets for Basque used in research, we also compiled two new sets. Nonetheless, as can be seen in the description below, the new sets are accurately described to allow easy identification. Secondly, to use automatic metrics, the texts must be available in Spanish and Basque, as a reference translation is necessary.

Linked to this, it was considered important to check if the source text was originally written in Spanish or whether it was already a translation, as this may introduce a degree of translationese in the original text. Similarly, it was considered important to track the original source language used to obtain the Basque translations, as this might, once again, introduce a degree of translationese. Also, it was important that the Basque translation was not a post-edited version, as this would bias the results in favour of the MT system that output a proposal closer to the original system used to create the reference.

When dealing with a minority language such as Basque, finding data sets that meet all the required criteria can prove challenging. We finally opted for the five data sets presented below; three publicly released data sets and two sets specifically created for this study. All five belong to different domains and have been translated into Basque without the aid of an MT system. Unfortunately, the original language of the texts and the source language for the translations was not always traceable.

Given the relatively large size of the corpora and the use restrictions of the MT systems (see subsection 3.1), we extracted 100 segments per set. Please remember that a segment might include, depending on the corpus, one or several sentences.

- **QTLeap IT Corpus.** The QTLeap corpus[7] consists of 4,000 question and answer pairs in the domain of computer and IT troubleshooting for both hardware and software. The text was gathered by an IT support company through their chat support service. As a result, the corpus consists of naturally occurring utterances produced by users while interacting with a service. Both Spanish and Basque are professional translations of the English text, which, in turn, is a professional translation of the original Portuguese text.

- **QTLeap News Corpus.** The QTLeap News corpus[8] is a sample of the News Commentary corpus created as training data resource for the Conference for Statistical Machine Translation Evaluation Campaign[9]. It consists of political and economic news crawled from the Project Syndicate site. The QTLeap News corpus consists of 1,104 sentences made available by the WMT 2012 and 2013 translation tasks, where the Spanish version was distributed (a manual translation of the original English text), and the Basque version was obtained through professional translation of the original English text.

- **TED Talks Corpus.** This data set was released by the 2018 IWSLT Evaluation for one of the two official translation tasks: Low Resource MT of TED talks from Basque to English Speech Translation of lectures, and is available at the Web Inventory Transcribed and Translated Talks[10] (Cettolo, Girardi, and Federico, 2012). We further cleaned the corpus to fix a number of alignment mismatches. The final set available consists of 6,649 parallel sentences. The corpus includes a range of miscellaneous talks given by distinguished experts. According to the TED Talks translation initiative, transcribed talks are translated by volunteers from their original language (mainly English) into the target languages. Therefore, both the Spanish and Basque translations would result from English source texts.

- **GuggenSet.** This set was compiled by extracting the text from the web page of Guggenheim Bilbao. It consists of the texts corresponding to the Essentials of the Collection section[11] where the main authors and their work are described. Therefore, the set belongs to the area of art, and formal and specialised register. Whereas the language of the original text is not specified, given the socio-linguistic context of the region, it is highly likely that this was Spanish and that the translation was carried out from Spanish to Basque by professional translators. This data set consists of 100 sentences.

- **AdminSet.** This set includes an extract of a law passed by the Basque Government.

[7] https://metashare.metanet4u.eu/

[8] https://metashare.metanet4u.eu/

[9] http://www.casmacat.eu/corpus/news-commentary.html

[10] https://wit3.fbk.eu/mt.php?release=2018-01

[11] https://www.guggenheim-bilbao.eus/en/the-collection

Data set	Domain	Source sentences	Avr. sent. length	Source words
QTLeap IT	IT	114	10 (min. 2 – max. 28)	1,249
QTLeap News	news	101	21 (min. 3 – max. 70)	2,127
TED Talks	miscellaneous	149	12 (min. 1 – max. 50)	1,832
GuggenSet	art	100	32 (min. 1 – max. 75)	3,274
AdminSet	law	133	33 (min. 1 – max. 233)	4,474

Table 1: Features of the data sets

Specifically, it contains the first 100 segments (133 sentences) of the Law 10/2019, of June 27, on Territorial Planning of Large Commercial Establishments (1). The set belongs to the area of administration and law, and highly formal and specialised register. The text was originally drafted in Spanish and professionally translated into Basque by the in-house translation service of the Basque Government.

Table 1 shows a summary of the main features of the data sets. As we can see, sets belong to a specific domain (IT, news, miscellaneous, art and law) and include between 100-150 sentences. As expected given their spontaneous nature, the QTLeap IT corpus and the Ted Talks corpus consist of shorter sentences, 10-12 words on average, whereas the specialized GuggenSet and AdminSet consist of considerably longer sentences, 32-33 words on average, with the AdminSet including sentences of up to 233 words. We can see that the total number of words included in the sets varies from 1,249 words for the QTLeap IT corpus, to 4,474 words for the AdminSet, the shortest and longest respectively.

4 Translationese experiments

In this section we compare the system outputs with respect the four translationese features, namely, simplification, normalisation, explicitation and interference, measured in the form of lexical variety, lexical density, length ratio and part-of-speech (PoS) sequence, respectively.

4.1 Lexical variety

Lexical variety indicates how rich the vocabulary used in a text is. As is standard in the field, we measure the difference in type/token ratio (TTR) as an approximation for this feature. In this study, we compare the TTRs of the system outputs. It is assumed that the lower the ratio, the more reduced the vocabulary in the translation is, and therefore, the use of the target language is poorer. This could mean that the translations lack precision and that lexical richness is not fully exploited. Therefore, a user should be inclined to use systems with higher type/token ratios.

A word of caution is in order here. This measure must be taken with caution when MT output is involved as the risk exists that an increased number of types is achieved through the incorrect translation of words.

Results in Table 2 show that the lexical variety in terms of type/token ratio is very similar for all systems, which tend to be within a range of 0.02 points. However, there are two systems that consistently score highest, namely, the EJ-EBMT system and Google Translator.

4.2 Lexical density

Lexical density intends to measure the amount of information present in the text. To that end, we calculate the ratio between the number of content words (nouns, adjectives, verbs and adverbs) and the total number of words in the texts. It is assumed that the higher the number of content word density, the higher the information transferred from the source text will be, which is, in principle, what we aim for.

Results are displayed in Table 2. These reveal that except for the QTLeap IT corpus, where EJ-RBMT scores the highest lexical density, Google Translator obtains the best scores across all data sets.

4.3 Length ratio

MT systems tend to produce sentences of a similar length to the source text because they often lack the capacity to distance the output from the source pattern. When this is the case, the length ratio tends to be low. Again, we should warn that significant differences in sentence length could also be explained by incorrect translation outputs such as incomplete translations, a feature that has been observed in NMT systems in particular.

	Text origin	**QTLeap IT**	**QTLeap News**	**TED talks**	**GuggenCorpus**	**AdminCorpus**
Lex. variety	Google	0.47522	**0.63087**	0.51355	**0.62278**	0.47308
	Modela	0.48206	0.61169	0.50295	0.59064	0.40965
	Batua	0.47144	0.61246	0.52962	0.60364	0.48930
	EJ-NMT	0.47203	0.62361	0.52583	0.61040	0.47638
	EJ-RBMT	**0.48866**	0.62801	**0.53732**	0.61320	**0.49466**
	Reference	0.43769	0.61413	0.50996	0.61165	0.45208
Lex. density	Google	0.86486	**0.84899**	**0.79267**	**0.83114**	**0.80833**
	Modela	0.85313	0.81768	0.76180	0.82054	0.76571
	Batua	0.84994	0.83740	0.77645	0.82107	0.79485
	EJ-NMT	0.84340	0.83803	0.76999	0.82438	0.79179
	EJ-RBMT	**0.87010**	0.84088	0.78592	0.82945	0.50605
	Reference	0.81256	0.83715	0.79188	0.82693	0.79800
Length ratio	Google	**0.01069**	0.05998	**0.00318**	0.05436	0.10815
	Modela	0.00345	0.09586	0.01784	0.05831	0.08619
	Batua	0.01908	0.02496	0.05242	0.00128	0.08527
	EJ-NMT	0.01677	0.03920	0.04912	**0.00103**	0.08690
	EJ-RBMT	0.06298	**0.00921**	0.06452	0.03770	**0.05400**
	Reference	0.10919	0.03398	0.00819	0.00383	0.00755
Perplexity	Google	4.75425	**4.91998**	5.22633	4.67304	4.26184
	Modela	**5.0867**	4.89808	**5.39553**	**4.87391**	**5.22066**
	Batua	4.91754	4.45909	5.08883	4.84463	4.73409
	EJ-NMT	4.72987	4.75369	4.92077	4.71945	4.75306
	EJ-RBMT	3.00027	4.4454	4.82782	4.34122	3.32085
	Reference	4.70072	4.61032	5.26604	4.99856	4.63802

Table 2: Results for the translationese features for the different data sets and MT systems where the best results for each data set are shown in bold

We calculate the absolute difference in length (words) between the source text and the translations, normalised by the length of the source text. Results in Table 2 show that for the QTLeap IT corpus and the TED Talks corpus, Google Translator outputs the lowest length ratios, whereas it is EJ-RBMT that obtains the best scores for the QTLeap News corpus and the AdminSet. EJ-NMT, closely followed by Batua, scores best for the GuggenSet.

4.4 PoS sequence

Interference aims to account for the source language patterns that are kept in a translation when these do not necessarily belong in naturally occurring text in the target language. Toral (2019) proposes to observe the PoS patterns as an indicator of interference and calculate the difference between the perplexities of the translation's PoS sequence when compared against a language model of PoS sequences in the source and target languages. As a low perplexity indicates that the translation is similar to the language model, the higher the difference in perplexity between the source and language model, the more the translation will differ from the original language and the closer it will be from the target model.

To train the language models, we first compiled the corpora for Basque and for Spanish, which included general text retrieved from the web, news and administrative texts, of 3 and 3.2 million sentences, respectively. We then used ixa-kat (Otegi et al., 2016) to annotate the Basque corpus and ixa-pipes (Agerri, Bermudez, and Rigau, 2014) to annotate the Spanish corpus. Both tools use the naf specification (Fokkens et al., 2014) for PoS classification, which consists of 9 tags and identify common nouns (N), proper nouns (R), adjectives, (G), verbs (V), prepositions (P), adverbs (A), conjunctions (C), determiners (D) and others (O). Finally, we built the models with Modified Kneser-Ney smoothing and no pruning, considering n-grams up to $n = 6$ but had to assign default parameters to singletons, even when they are not present in the PoS-annotated corpus.

The results in Table 2 show that the translations with the lowest interference from the source are provided by Modela for all data sets, even when other systems are close to its results. Interestingly, the EJ-RBMT output is the system displaying the highest level of similarity toward the source language and

	QTLeap IT		QTLeap News		GuggenCorpus		TED Talks		AdminCorpus	
MT system	**BLEU**	**TER**	**BLEU**	**TER**	**BLEU**	**TER**	**BLEU**	**TER**	**BLEU**	**TER**
Google	**18.75**	**63.05**	12.58	71.90	15.16	67.01	**36.18**	**41.85**	19.05	59.63
Modela	16.09	69.07	13.46	**69.07**	**28.64**	72.53	17.71	67.15	17.20	70.10
Batua	17.55	64.77	**17.33**	69.74	21.93	62.62	29.11	47.36	22.31	58.81
EJ NMT	18.43	64.43	16.60	69.58	22.14	**61.66**	29.11	48.56	**23.39**	**58.36**
EJ RBMT	10.16	82.30	07.46	79.38	10.23	76.15	19.05	59.63	07.50	76.79

Table 3: Automatic metric results

distances most from the naturally occurring pattern of the target language.

5 *Automatic quality experiments*

This section analyses the output of the five MT systems under study when measured by automatic metrics. Specifically, we calculated BLEU and TER scores. Whereas they are both string-based, precision-oriented metrics that compare the MT output against a reference translation, it could be argued that BLEU is more directed to providing an overall quality measurement whereas TER bears more relationship with the editing work required by the output.

Results from Table 3 show interesting trends. We observe that systems tend to follow the same trend in variation when moving from one domain to the next, that is, in general, the quality of the output either increases or worsens for the whole group of systems. We can observe that, on average, systems perform worse on the QTLeap IT and QTLeap News corpora. The AdminSet obtains better results, followed closely by the GuggenSet. Overall, the set with best scores is the TED Talks corpus.

However, there are two aspects to highlight here. Firstly, we can observe that for each domain or data set, it is a specific MT system that stands out as best. For example, Google obtains the highest BLEU score for the QTLeap IT corpus and TED Talks corpus, while Batua performs best for QTLeap News corpus, Modela for the GuggenSet and EJ-NMT for the AdminSet. In turn, the RBMT system lags behind for all domains. Note that the TER scores point at different systems for three out of the five data sets.

Secondly, it is interesting to see that systems not only vary in output quality in unison, they also display BLEU scores within the same ranges for each domain, except at times in the specific sets in which each system stands out or lags behind. The NMT systems revolve around the range of 17.18 BLEU points for the QTLeap IT corpus, around 13-17 for the QTLeap News corpus, around 22-28 for the GuggenSet, around 29-36 for the TED Talks, and around 18-23 for the AdminSet. TER scores, in turn, also remain in ranges that vary from 2 to 11 points.

From the scores obtained, we could conclude that TED texts within the TED Talk style are best translated by Google Translate, administrative texts are best handled by EJ-NMT, formal art-related texts will require fewer edits if translated by EJ-NMT (even when the overall quality might be better with Modela), news are best translated with Batua, and Google Translate performs best with IT-related material. Needless to say that these conclusions must be approached with caution for two main reasons: (1) our test sets are small; and (2) these systems are not static but rather they are being improved and retrained with additional data and by using different techniques.

The limitations mentioned above, however, do not prevent us from comparing automatic metric results and translationese features. Firstly, we observe that it is not the same system that consistently performs best with regards translationese. This raises the question of what the implications for post-editing each of the features are and which could be more or less relevant for users. Further analyses of post-editing effort would be necessary to establish the impact each translationese feature has on MT output use.

Secondly, by comparing the rankings assigned by the different features and metrics, we observe that automatic metrics and translationese do not always point at the same MT system as the best performer. The same question as above emerges here, that is, which type of information is more useful for a user who wants to post-edit the output? A closer look at the rankings proposed by BLEU and TER reveals that, whereas different between them, none of them is in more agreement with translationese results.

Interestingly although not unexpectedly, it is worth noting the presence of the EJ-RBMT system in high positions of the ranking by certain translationese features, whereas automatic metrics consistently show that its output quality is considerably poorer.

6 Conclusions

It is believed that translations display a set of shared features that distinguishes them from texts written in the original language, referred to as Translation Universals, which results in translationese. In this work, we set to explore the possibility of using such features as indicators of MT quality for users to select an MT system for post-editing assuming that a lower level of translationese will reveal a reduced need for editing. To that end, we compared the results obtained from translationese features and automatic metrics for five data sets and MT systems. Whereas further experiments using large data sets and variations of the approaches to measure the features should be performed to gather conclusive data, and contrast these results with user post-editing performance, results seem to indicate that the two sets of metrics rank systems differently, opening an avenue for research into the information each provides.

Acknowledgements

This research was partially supported by the Spanish MEIC and MCIU (UnsupNMT TIN2017-91692-EXP and DOMINO PGC2018-102041-B-I00, co-funded by EU FEDER), and the BigKnowledge project (BBVA foundation grant 2018).

Bibliography

Agerri, R., J. Bermudez, and G. Rigau. 2014. Ixa pipeline: Efficient and ready to use multilingual nlp tools. In *LREC*, volume 2014, pages 3823–3828.

Baker, M. 1993. Corpus linguistics and translation studies: Implications and applications. *Text and technology: In honour of John Sinclair*, 233:250.

Baroni, M. and S. Bernardini. 2005. A new approach to the study of translationese: Machine-learning the difference between original and translated text. *Literary and Linguistic Computing*, 21(3):259–274.

Cettolo, M., C. Girardi, and M. Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the EAMT*, pages 261–268, Trento, Italy, May.

Etchegoyhen, T., E. Martínez Garcia, A. Azpeitia, G. Labaka, I. Alegria, I. Cortes Etxabe, A. Jauregi Carrera, I. Ellakuria Santos, M. Martin, and E. Calonge. 2018. Neural machine translation of basque. In *Proceedings of the 21st Annual Conference of the EAMT, 28-30 May, Alacant, Spain*, pages 139–148.

Fokkens, A., A. Soroa, Z. Beloki, N. Ockeloen, G. Rigau, W. R. Van Hage, and P. Vossen. 2014. Naf and gaf: Linking linguistic annotations. In *Proceedings 10th Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*, pages 9–16.

Green, S., J. Heer, and C. D. Manning. 2013. The efficacy of human post-editing for language translation. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 439–448. ACM.

Laviosa, S. 2002. *Corpus-based translation studies: theory, findings, applications*, volume 17. Rodopi.

Otegi, A., N. Ezeiza, I. Goenaga, and G. Labaka. 2016. A Modular Chain of NLP Tools for Basque. In *Proceedings of the 19th International Conference of Text, Speech, and Dialogue, Brno, Czech Republic, September 12-16.* pages 93–100.

Tirkkonen-Condit, S. 2002. Translationese - a myth or an empirical fact? a study into the linguistic identifiability of translated language. *Target. International Journal of Translation Studies*, 14(2):207–220.

Toral, A. 2019. Post-editese: an exacerbated translationese. In *Proceedings of MT Summit XVII, 19-23 August, Dublin, Ireland*, pages 273–281.

Toury, G. 2012. *Descriptive translation studies and beyond: Revised edition*, volume 100. John Benjamins Publishing.

Volansky, V., N. Ordan, and S. Wintner. 2013. On the features of translationese. *Digital Scholarship in the Humanities*, 30(1):98–118.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 101-108 recibido 01-12-2019 revisado 14-01-2020 aceptado 16-01-2020

Uso de técnicas basadas en one-shot learning para la identificación del locutor

Speaker Identification using techniques based on one-shot learning

Juan Chica, Christian Salamea
Universidad Politécnica Salesiana
Grupo de Investigación en Interacción, Robótica y Automática
jchicao@ups.edu.ec, csalamea@ups.edu.ec

Resumen: Un sistema para la identificación de locutor, para ser eficaz requiere una extensa cantidad de muestras de audio por cada locutor que no siempre es fácil de obtener. En contraste, sistemas basados en Meta-learning (en español, aprender a aprender) como one-shot learning utilizan una única muestra para diferenciar entre clases. En este trabajo se evalúa el potencial de un sistema de meta-learning para la identificación del locutor independiente del texto. En la experimentación se utilizan: espectrograma de mel, i-vectores y re muestreo (downsampling) para procesar el audio y obtener un vector de características. Este vector es la entrada de una red neuronal siamesa que se encarga de realizar la identificación. El mejor resultado se obtuvo al diferenciar entre 4 locutores con una exactitud de 0.9. Los resultados mostraron que el uso de técnicas basadas en one-shot learning tiene gran potencial para ser usados en la identificación del locutor y podrían ser muy útiles en ambientes reales como la biometría o ámbitos forenses por su versatilidad.

Palabras clave: Identificación del locutor, Independiente de texto, Meta Learning, N-way clasification, One-Shot learning, Redes Neuronales Siamesas, Voxceleb1

Abstract: A speaker identification system in order to be effective requires a large number of audio samples of each speaker, which are not always accessible or easy to collect. In contrast, systems based on meta-learning like one-shot learning, use a single sample to differentiate between classes. This work evaluates the potential of applying the meta-learning approach to text-independent speaker identification tasks. In the experimentation mel spectrogram, i-vectors and resample (downsampling) are used to both process the audio signal and to obtain a feature vector. This feature vector is the input of a siamese neural network that is responsible for performing the identification task. The best result was obtained by differentiating between 4 speakers with an accuracy of 0.9. The obtained results show that one-shot learning approaches have great potential to be used speaker identification and could be very useful in a real field like biometrics or forensic because of its versatility.

Keywords: Speaker Identification, Text independent, Meta Learning, N-Way clasification, One-Shot learning, Siamese Neural Network, Voxceleb1

1 Introducción

La voz en los seres humanos es producida a través de un largo proceso que se inicia desde el momento en el que se aspira aire, el cual va hacia los pulmones, pasando por la caja torácica y termina en el tracto vocal que es donde se articula la voz. Durante este proceso, la resonancia (formantes) que se genera en el tracto vocal queda registrada el contenido en frecuencia de la señal de la voz. Por otra que, al analizar la forma del espectro de la señal de voz es posible estimar la forma del tracto vocal de una persona y por consiguiente reconocer la identidad de un individuo particular. En el campo de la biometría este tipo de tarea es conocida como reconocimiento de locutor que

ISSN 1135-5948. DOI 10.26342/2020-64-12

incluye la verificación y la identificación de una persona por su voz.

La ventaja de utilizar la señal de la voz para tareas de reconocimiento de locutor recae en el hecho de que la voz es una señal natural emitida por las personas, por lo que la obtención de esta señal no se considera peligrosa ni invasiva (desde un punto de vista físico). Sin embargo, se debe considerar que la señal de audio que se obtiene en situaciones reales puede contener ruido ambiental (sonidos externos del entorno), siendo necesario el utilizar alguna técnica para que en una muestra de audio únicamente se tenga la información de interés (voz), o en su defecto que se enfatice dicha señal de interés. En los sistemas actuales de reconocimiento es necesario utilizar grandes cantidades de información con la finalidad de obtener un buen rendimiento, no obstante, nuevas técnicas desarrolladas basadas en Meta Learning (aprender a aprender), permiten disminuir la cantidad de datos necesarios para obtener un buen resultado.

En este trabajo se propone el uso de un sistema compuesto por dos etapas un Front-End y un Back-End para la identificación del locutor de texto independiente. En el Front End, se realiza un procesamiento de las señales de audio para extraer sus características más relevantes, aquí se utilizan tres enfoques diferenciados por el tipo de procesamiento realizado en la señal de audio. En el primer enfoque, se realiza un re-muestreo de la señal de audio, la cual pasa de 16000 Hercios (Hz) a 1000Hz de frecuencia de muestreo. En el segundo enfoque, se utiliza una técnica basada en coeficientes cepstrales y que ha sido ampliamente usada para el procesamiento del habla, esta técnica se denomina i-vectores y consiste en obtener un vector de características que permite representar la señal original que se encuentra en una alta dimensión, en un formato de baja dimensión sin perder las características de la alta. Por último, en el tercer enfoque se obtiene el espectrograma de mel en escala logarítmica de las señales de audio. La salida de cada enfoque se utiliza como entrada en el Back-End, donde se determina cual es el más óptimo para ser utilizado en el sistema. Por otra parte, en el Back-End se realiza la tarea de clasificación o de identificación al locutor como tal utilizando un modelo de redes neuronales siamesas que a su salida da como resultado una medida de similitud entre dos observaciones.

El objetivo de este trabajo es evaluar el potencial uso de técnicas basadas en one-shot learning en tareas de identificación de locutor de texto independiente. Nos hemos decantado por el uso de esta técnica debido a que una vez se ha entrenado el modelo, se puede identificar locutores que incluso no han sido presentados al modelo durante la fase de entrenamiento, lo cual es una gran ventaja en aplicaciones reales. Por otra parte, para evaluar el sistema en un ambiente real se seleccionó la base de datos de Voxceleb1, ya que contiene audios grabados en condiciones reales (diferentes ambientes) y las personas están utilizando su lenguaje natural (independencia del texto). Los resultados obtenidos muestran que este tipo de técnicas tienen gran potencial para ser utilizadas en la identificación de locutores, particularmente el mejor rendimiento se alcanza al diferenciar un locutor de entre 4 pues se obtiene alrededor de 0.9 (90%) de exactitud al identificarlo.

2 *Fundamento Teoríco*

2.1 Estado del Arte

Los primeros trabajos sobre reconocimiento de locutor se remontan a los años 50s, donde se ejecutaron los primeros intentos por construir una máquina para tareas de reconocimiento en base a los principios fundamentales de la fonética. En 1952, Davis, Biggulph y Balashek desarrollaron el primer sistema documentado que permitía identificar el habla aislada pronunciada por un solo locutor. Este sistema estaba basado en filtros analógicos que extraían medidas de las resonancias espectrales de tracto vocal por cada digito. En la década de los 70s las variaciones intra-hablantes comienzan a ser investigadas por Endres (Endres et al., 1971) et.al. y Furui (Furui, 1981). Siendo este último quien propuso usar los coeficientes espectrales como características fundamentales, aunque no fueron consideradas relevantes en principio, años después a finales de los 80 aumentó su uso e incluso continúan siendo utilizadas en la actualidad en casi todos los sistemas de reconocimiento de voz (Donoso García del Castillo, 2014). Durante la época de los 80s, en los sistemas de reconocimiento de habla se comenzaron a utilizar técnicas de programación dinámica, pero fueron reemplazadas posteriormente por los modelos ocultos de Markov (HMM del inglés Hidden Markov Models), y a su vez, por su buen rendimiento en estas tareas también fueron utilizados en los primeros sistemas de reconocimiento de locutor (Univaso, 2017). Sin embargo, no fue sino hasta principios de los 90s, que se introduce el concepto del modelo de Mezclas Gaussianas (GMM) el cual luego fue empleado en la mayoría de los sistemas de reconocimiento. Por otra parte, en esta misma época también se profundizó en los sistemas robustos a variables no deseadas como el **ruido**, las diferencias de canal y la variabilidad inter locutor. A finales de esta década, se inician evaluaciones de sistemas de reconocimiento de locutor (SRE) llevadas a cabo por el Instituto Nacional de Estándares y Tecnología (NIST) que buscan determinar el estado del arte, y continúan hasta la actualidad.

En los años 2000, se comienzan a introducir diversas técnicas de normalización en los modelos y también se inicia el uso de máquinas de soporte vectorial (SVM) como clasificadores (Wan and Campbell, 2000). Finalmente, en la década del 2010 es donde se han dado grandes avances y se han propuesto técnicas que son investigadas y utilizadas en la actualidad. A principios de esta época, en busca de remover o atenuar las características que no son propias del hablante, se inician los estudios sobre la representación del habla en subespacios vectoriales. Así, se plantean técnicas como el análisis factorial conjunto (Joint Factor Analysis, JFA) (Kenny, 2006) o técnicas basadas en el espacio de variabilidad total como los i-vectores (Dehak et al., 2011), que son consideradas el estado del arte según las evaluaciones de NIST (Univaso, 2017).

2.2 One-Shot Learning

En tareas tradicionales de clasificación, a partir de una observación (representada por un vector de características) un modelo obtiene a su salida la probabilidad de que esta observación pertenezca a una clase particular. Por ejemplo, si se tiene un modelo entrenado y se quiere determinar si una imagen particular es de un gato, un perro u otro animal, el modelo genera una salida con 3 valores (correspondientes a cada una de las 3 clases), siendo el mayor de ellos el correspondiente a la clase más probable. En este tipo de sistemas resulta complejo añadir una nueva clase cuando un modelo ya ha sido entrenado, pues se requiere grandes cantidades de observaciones de cada clase para aprender sus características particulares y poder diferenciarlas. En contraste, técnicas basadas en el concepto de one-shot learning (utilizar una única observación para diferenciar una clase) pueden mejorar la flexibilidad de las técnicas tradicionales de clasificación (Li Fe-Fei et al., 2003). En este tipo de técnicas, en lugar de evaluar directamente una observación para determinar a qué clase pertenece (como en un sistema de clasificación tradicional), se toma una observación adicional que sirve como referencia para obtener un valor de similitud entre esta y una nueva observación que se presente, siendo esta última de la que se quiere determinar a qué clase pertenece. Algo importante a notar es que, en este caso el modelo no aprende directamente a clasificar, sino que está aprendiendo a determinar qué tan similares son dos observaciones (1 similar, 0 no hay similitud) y a través de esto clasificar.

Existen diferentes arquitecturas que permiten realizar tareas de clasificación utilizando one-shot learning (Koch et al., n.d.; Li Fe-Fei et al., 2003; Santoro et al., 2016; Sung et al., 2018; Vinyals et al., 2016). Entre ellas, el método basado en redes siamesas (siamese neural networks) ha sido uno de los que ha obtenido muy buenos resultados en tareas de clasificación. En la figura 1 se muestra un diagrama general de una red siamesa, se puede observar que está formada por dos redes neuronales convolucionales (CNN) las cuales no son diferentes, sino que son dos copias de la misma red, básicamente comparten sus parámetros (son dos redes gemelas de ahí la terminología de siamesas). Las entradas son una observación de referencia de la clase denominada “Anchor” y una observación de entrada de la cual se quiere determinar a qué clase pertenece. Ambas pasan a través de las capas convolucionales y suponiendo que el modelo fue entrenado correctamente se tendrán dos casos posibles: si las dos entradas pertenecen a la misma clase, su vector de características (encoding) debe ser similar, mientras que, si son de una clase diferente, su vector de características será diferente. De igual forma, si se obtiene la diferencia absoluta (Absolute Difference) entre los dos vectores de características, este resultado será diferente dependiendo del caso que se presente. Finalmente, el resultado de la diferencia de los vectores, ingresa a una última capa que contiene únicamente una función sigmoide que da a su salida un valor de 1 si los vectores son similares y un valor de 0 si son diferentes.

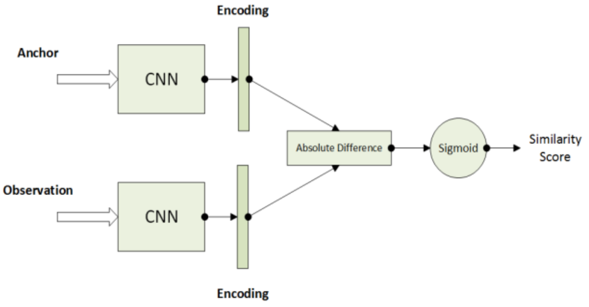


Fig. 1. Arquitectura general de una red siamesa

Se puede notar que tal como en el concepto de one-shot learning, utilizando redes siamesas lo que se busca es aprender a clasificar por medio de una función de similitud, es decir lo que el modelo aprende es la **tarea** de distinguir la similitud más no a clasificar entre clases directamente.

3 Metodología Experimental

3.1 Sistema Propuesto

En este trabajo se propone el uso de un sistema compuesto por un Front-End que se encarga del procesamiento de la señal de audio y un Back-End donde se realiza la tarea de la identificación del locutor como tal. Por otra parte, para evaluar el sistema se seleccionó la base de datos de Voxceleb1 y la división realizada en la misma; siendo 1211 locutores para entrenamiento y 40 locutores para testeo. Además, debido a que en la base de datos existían audios de diferentes duraciones se utilizaron únicamente 20 audios de 5 segundos de duración por cada locutor (los audios de menor duración se descartaron y los de mayor duración se recortaron a 5 segundos), con la finalidad de valorar la capacidad del sistema para aprender utilizando una menor cantidad de datos que sistemas tradicionales.

Partiendo de la base de datos, en el Front-End, se utilizan tres enfoques diferentes para disminuir el tamaño del vector de entrada y a su vez adecuar la señal de audio acorde a la arquitectura del modelo implementado en el Back-End. En el primer enfoque, el audio fue re-muestreado (downsampling) a una frecuencia mucho menor; de 16000 Hz se pasó a 1000 Hz y luego este vector unidimensional se transformó en un vector bidimensional de 70x70, el cual es la entrada del Back-End. Cabe destacar que, únicamente en este enfoque se elimina un segundo de la muestra para regularizar el tamaño del vector (al re-muestrear la señal se redondea hacia abajo quedando un vector bidimensional de 70x70) y esta metodología se acopla al objetivo del primer enfoque pues se busca evaluar la capacidad del sistema para identificar un locutor cuando existe pérdida de información. En el segundo enfoque, se utilizaron los i-vectores de las señales de audio. Para ello, en primer lugar, se obtuvieron los coeficientes cepstrales en escala de mel (MFCC) de los audios utilizando un tamaño de ventana de 25 milisegundos (ms), espaciado cada 10 ms y 26 filtros triangulares. Luego, utilizando estos MFCC se entrenó un UBM (Universal Background Model) el cual fue utilizado para obtener 400, 900 y 1600 i-vectores. De igual manera que en el primer enfoque, el vector que contiene los i-vectores se concatena y se transforma en un vector bidimensional (de tamaño 20x20, 30x30 y 40x40, respectivamente) que es la entrada del Back-End. Cabe señalar que todo el proceso realizado en este enfoque se lo hizo utilizando la librería de Sidekit (“Welcome to SIDEKIT 1.3.1 documentation! — SIDEKIT documentation,” n.d.). Finalmente, en el tercer enfoque de los audio se extrae el espectrograma de mel en escala logarítmica dado que se ha demostrado que el mismo obtiene una muy buena representación de una señal de audio (Choi et al., 2018). El espectrograma se obtuvo utilizando 67 filtros de mel generando así un vector bidimensional de dimensiones 67x67, el cual, es la entrada del Back-End. En la figura 2 se puede observar el diagrama general de funcionamiento del sistema propuesto.

Por otra parte, para la identificación del locutor en el Back-End se utiliza una red neuronal siamesa que está basada en la arquitectura y metodología propuesta por Kotch et al. (Koch et al., n.d.). El modelo usado está compuesto por dos redes neuronales convolucionales (ConvNet) las cuales no son diferentes, sino que comparten la misma estructura y valores de parámetros (luego de cada ConvNet se realizó normalización del lote o batch). De esta forma, a la entrada de estas, se

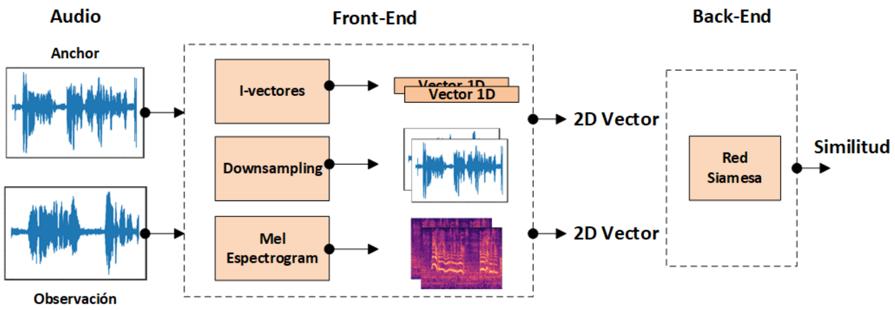


Fig. 2. Diagrama general de funcionamiento del Sistema Front-End Back-End

tendrán dos vectores (uno de referencia o anchor y una observación) los cuales pasan a través de la ConvNet y dan como resultado un vector de características de longitud fija por cada entrada. Ahora, si comparamos estos dos vectores pueden existir dos casos; si los dos audios pertenecen al mismo locutor sus vectores de características deben ser similares y si no son del mismo locutor deberían ser diferentes. Por tanto, de la salida de la red siamesa se calcula la diferencia absoluta (L1 distance) de los dos vectores de características que se obtiene de la ConvNet y este vector resultante es la entrada de una última capa densa de neuronas (fully-conected) que tiene una función de activación de tipo sigmoide, quien a su salida entregara un valor (1) en el caso de que las entradas sean del mismo locutor y otro valor (0) en el caso de que sean de un locutor diferente. De esta forma, durante el entrenamiento lo que se busca es que el modelo adapte sus parámetros de tal forma que pueda ser capaz de mejorar su exactitud al identificar si un par de observaciones pertenecen o no al mismo locutor. En la figura 3 se muestra la arquitectura de la red siamesa utilizada.

3.2 Métrica de Evaluación

En la tarea de identificación de locutor la idea principal es determinar a qué locutor pertenece la voz de entre varios locutores, por lo que este tipo de problemas podría ser catalogado como una clasificación multiclase. Sin embargo, en one-shot learning y con la red neuronal siamesa lo que se busca no es que el modelo aprenda a diferenciar entre clases, sino que aprenda a extraer características y que determine una medida de similitud entre dos observaciones. Así, la clasificación multiclase se transforma en una tarea de clasificación binaria donde una clase representa una similitud total y otra clase que no existe similitud. En este contexto, para evaluar el rendimiento de cada enfoque se tomó una estrategia denominada N-way One-shot clasification [9], donde N representa la cantidad de pares de observaciones a evaluar.

Por ejemplo, en una estrategia 4-way one-shot clasification se tendría una observación base (audio a identificar a cuál locutor pertenece) y se compararía con respecto a otras 4 observaciones donde únicamente una es del mismo locutor, por lo que se espera que el modelo obtenga una métrica de similitud alta a esta observación y más baja para las otras observaciones que son de otros locutores. Así, si entre las dos observaciones que son del mismo locutor se obtiene la similitud máxima la clasificación es correcta, caso contrario, se la consideraría incorrecta. En la figura 4 se ilustra una clasificación correcta en un ejemplo de 4-way one-shot clasification. Si estas pruebas son realizadas m cantidad de veces, se puede obtener un resultado más general del rendimiento del enfoque, por lo que, la métrica final de evaluación de los enfoques está en función de:

$$S = \frac{mc}{m}$$

Donde mc representa cuantas clasificaciones se realizaron correctamente durante m cantidad de veces y S representa su exactitud.

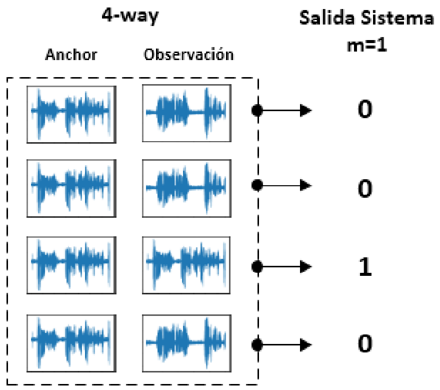


Fig. 4. Ilustración de una evaluación basada en N-way Clasification para N=4

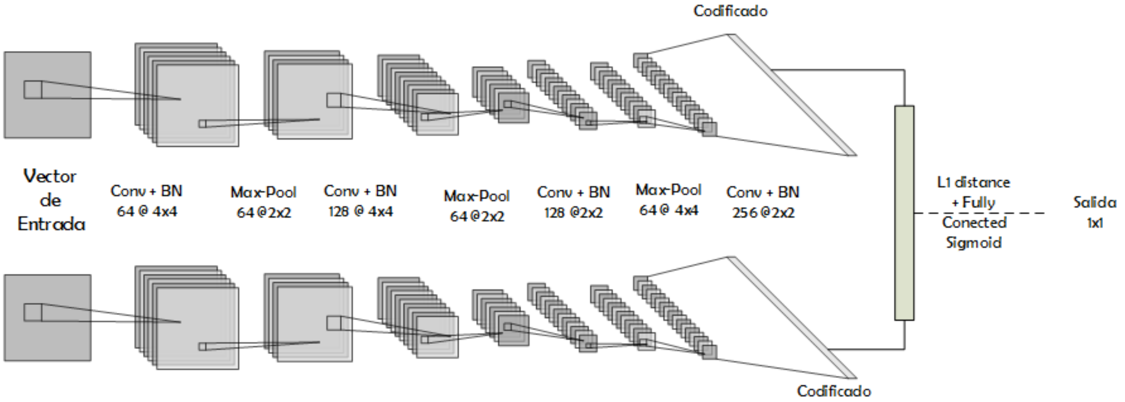


Fig. 3. Arquitectura de la red neuronal siamesa implementada en el Back-End

Si todo se clasificó correctamente, se obtendría un valor de S=1, cada vez que se cometa un error este valor ira disminuyendo hasta alcanzar 0 que sería el caso en que ninguna clasificación fue realizada correctamente.

3.3 Fase de Entrenamiento

En el entrenamiento del sistema se utilizaron 80 épocas (aproximadamente 20000 iteraciones), un tamaño de batch de 100, la función de perdida de entropía cruzada binaria (binary cross-entropy) y el algoritmo de Adam con una tasa de aprendizaje de 0.00001 (para tasas de mayor valor se volvía inestable). En las pruebas realizadas, el enfoque basado en i-vectores fue el que peor rendimiento obtuvo, como se puede observar en la figura 5 incluso para diferenciar entre 4 locutores (N=4) no se alcanza un buen resultado y al aumentar el número de locutores a diferenciar su exactitud fue disminuyendo aún más. Este comportamiento posiblemente se deba a que, la información que está presente en los i-vectores se vuelve dispersa con respecto al audio original y causa que en las capas convolucionales no sea posible extraer información relevante para medir la similitud.

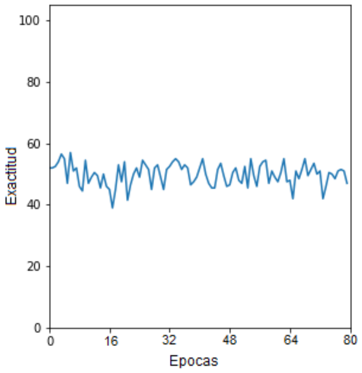


Fig. 5. Tendencia de la exactitud para tamaño del batch de 100 y N=4 en enfoque i-vectores

Por otra parte, en los otros enfoques ocurre un comportamiento opuesto al mostrado con i-vectores, en ambos casos se puede notar que si existe una tendencia de ascendente en la exactitud. En la figura 6, se muestra la tendencia de la exactitud en enfoque de downsampling durante el entrenamiento, para N=4. Mientras que en la figura 7, se muestra la tendencia del enfoque del espectrograma de mel, pero para N=7.

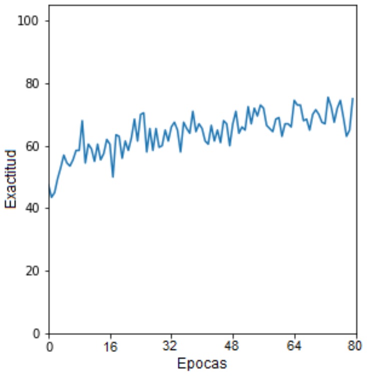


Figura 6. Tendencia de la exactitud para tamaño del batch de 100 y N=4 en enfoque Downsampling

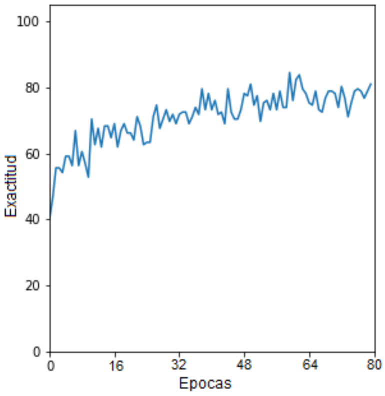


Figura. 7. Tendencia de la exactitud para tamaño del batch de 100 y N=7 en enfoque Espectro de Mel

Al comparar ambas figuras se puede notar la diferencia que existe en los dos métodos pues, aunque en el caso del espectrograma de mel se está utilizando una mayor cantidad de locutores (N mayor), supera al rendimiento de downsampling que utiliza menos locutores, se alcanzó una exactitud de 0.84 y 0.75, respectivamente.

4 *Resultados*

Los experimentos realizados tienen la finalidad de identificar el potencial que tienen estos tipos de sistemas para tareas de identificación del locutor, por lo que la métrica sobre la exactitud, se obtiene con respecto a un grupo de locutores diferentes a los locutores utilizados en el entrenamiento (grupo de testeo) y utilizando m=100. Es decir, la métrica mostrada es el promedio de 100 pruebas de testeo realizadas para esa observación. Las diferentes pruebas se realizaron modificando el valor de N, siendo el menor valor N=4 y el valor más alto utilizado N=40. En las pruebas realizadas para N=40, es decir para diferenciar a un locutor particular de entre otros 39 locutores, la exactitud disminuyó considerablemente con respecto a lo alcanzado para un menor valor de N. Por lo que, aunque se fijó el valor de 80 épocas para todos los experimentos, dado el bajo rendimiento y que el número de N es el mayor de todas las pruebas, se realizó una prueba adicional con el doble de épocas, es decir para 160 épocas en total.

Por otra parte, de los tres enfoques, el basado en i-vectores fue para el que peor rendimiento obtuvo en todas las pruebas, por lo que, no se muestran los resultados con respecto a i-vectores por su bajo rendimiento (para N=4 se alcanzó una exactitud de 54% y para valores mayores de N su exactitud fue menor) en comparación a los otros enfoques. En la tabla 1 se muestran los resultados de los enfoques basados en el espectrograma de mel y downsampling, se puede observar que para un valor de N=4 el rendimiento alcanza un S=0.9 para el enfoque del espectrograma de mel mientras que para downsampling se alcanza un S=0.75. Sin embargo, conforme se aumenta la cantidad de locutores a evaluar (N>4), la exactitud disminuye para ambos enfoques. En las pruebas adicionales realizadas para N=40 con el doble de épocas (160) se obtuvo una mejoría para ambos enfoques alcanzando un 66% de exactitud con el espectrograma de mel y 45% para downsampling, es decir una mejoría del 10% con respecto a las pruebas con 80 épocas.

Enfoque (Front-End)	N (Locutores)	Exactitud (m=100)
Downsampling	4	75 %
	7	61 %
	10	54 %
	20	47 %
	40	35 %
Espectrograma de Mel	4	90 %
	7	84 %
	10	76 %
	20	67 %
	40	56 %

Tabla 1: Resultados Obtenidos en la Fase de Experimentos para 160 Épocas

5 *Conclusiones*

En este trabajo se propone el uso de un sistema basado en one-shot learning para la identificación del locutor y se evalúa su funcionamiento utilizando audios que contienen tanto la voz de un locutor como ruido ambiental, para valorar su rendimiento en un ambiente real. En los experimentos realizados se pudo notar que, a medida que se aumentaba la cantidad de locutores a diferenciar, la exactitud disminuye hasta alcanzar un 66% al diferenciar de entre 40 locutores a la vez, para una cantidad mayor de locutores, considerando el comportamiento del sistema durante los experimentos el rendimiento podría ser aún menor. Se piensa que este comportamiento podría deberse a que, al aumentar la cantidad de locutores, es probable que varios de ellos tengan características similares en su voz causando que la información se solape y se complique la tarea de diferenciarlos. Por lo que en trabajos futuros se tomara en consideración el método usado en el Front-End, para evitar perder información que podría ser determinante a la hora de diferenciar dos audios con características muy similares en la voz. Sin embargo, cabe señalar que, incluso al utilizar Downsampling de una forma tan agresiva el sistema obtuvo un resultado aceptable, de esta manera se pudo constatar la capacidad que se tiene para identificar un locutor, aun cuando existe gran pérdida de información con respecto al audio original.

De entre todas las arquitecturas propuestas, la que mejor rendimiento obtuvo fue la que utiliza el espectrograma de mel en escala de la señal de audio original para determinar a quién pertenece la voz a través de una medida de similitud entre dos observaciones. Cabe destacar que, en este tipo de sistemas para la identificación se utiliza únicamente una muestra de audio y además existe la posibilidad de utilizar la voz de locutores diferentes a los presentados en el entrenamiento (algo que no es posible en un sistema de clasificación tradicional). Estas características, le brindan al sistema gran escalabilidad para ser utilizado en situaciones reales donde puede ser complicado adquirir gran cantidad de muestras de un locutor. Por lo que, según los resultados obtenidos y las ventajas que presentan estos sistemas durante el entrenamiento y el funcionamiento, los sistemas basados en one-shot learning tienen gran potencial para ser utilizados en tareas de identificación del locutor independiente del texto.

Bibliografía

Endres, W., W. Bambach, G. Flösser, 1971. Voice spectrograms as a function of age, voice disguise, and voice imitation. *Journal Acoustic Society of America*. Volumen:49, páginas 1842–1848.

Furui, S., 1981. Cepstral analysis technique for automatic speaker verification. *IEEE Trans. on Acoustics, Speech and Signal Processing*.Volumen:29, páginas 254–272.

Donoso, R., 2014. Diseño e implementación de un sistema de reconocimiento de hablantes. *Universidad Carlos III de Madrid.*

Univaso, P., 2017. Forensic speaker identification: a tutorial. *IEEE Latin America Transaction.* Volumen:15, páginas 1754–1770.

Wan, V., W. Campbell, 2000. Support vector machines for speaker verification and identification. *Neural Networks for Signal Processing X.* Volume:2, páginas 775–784.

Kenny, P., 2006. Joint Factor Analysis of Speaker and Session Variability: Theory and Algorithms. *Computer Science.*

Dehak, N., P. Kenny, R. Dehak, P. Dumouchel, P. Ouellet, 2011. Front-End Factor Analysis for Speaker Verification. *IEEE Trans. Audio Speech Lang. Proces.* Voumen:19, páginas788–798.

Fe-Fei, L., R. Fergus, P. Perona, 2003. A Bayesian approach to unsupervised one-shot learning of object categories. *Proceedings Ninth IEEE International Conference on Computer Vision.* Volumen:2,páginas 1134–1141, France.

Koch, G., R. Zemel, R. Salakhutdinov, 2015. Siamese neural networks for one-shot image recognition. *Proc. 32 International Conference on Machine Learning.* Volumen:37, France.

Santoro, A., S. Bartunov, M. Botvinick, D. Wierstra, T. Lillicrap, 2016. Meta-Learning with Memory-Augmented Neural Networks. *Proc. 33 International Conf. on Machine Learning*, USA.

Sung, F., Y. Yang, L. Zhang, T. Xiang, P. Torr, T. Hospedales, 2018. Learning to Compare: Relation Network for Few-Shot Learning. *IEEE Conference on Computer Vision and Pattern Recognition.* Páginas 1199–1208, USA

Vinyals, O., C. Blundell, T. Lillicrap, K. Kavukcuoglu, D. Wierstra, 2016. Matching Networks for One Shot Learning. *30th Conference on Neural Information Processing Systems.* páginas 3630–3638, España.

SIDEKIT 1.3.1 documentation, URL https://projets-lium.univ-lemans.fr/sidekit/ (accessed 9.5.19).

Choi, K., G. Fazekas, M. Sandler, K. Cho, 2018. A Comparison of Audio Signal Preprocessing Methods for Deep Neural Networks on Music Tagging. *26th Eur. Signal Process Conference,* páginas 1870–1874, Italy.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 109-116 recibido 02-12-2019 revisado 09-01-2020 aceptado 11-01-2020

Sentiment Analysis in Spanish Tweets: Some Experiments with Focus on Neutral Tweets

Análisis de Sentimiento para Tweets en Español: Algunos Experimentos con Foco en los Tweets Neutros

Luis Chiruzzo, Mathias Etcheverry, and Aiala Rosá
Universidad de la República, Montevideo, Uruguay
{luischir,mathiase,aialar}@fing.edu.uy

Abstract: We present different methods for Sentiment analysis in Spanish tweets: SVM based on word embeddings centroid for the tweet, CNN and LSTM. We analyze the results obtained using the corpora from the TASS sentiment analysis challenge, obtaining state of the art results in the performance of the classifiers. As the neutral category is the hardest one to classify, we focus in understanding the neutral tweets classification problems and we further analyze the composition of this class in order to extract insights on how to improve the classifiers.
Keywords: Sentiment Analysis, Spanish Tweets Analysis, Machine Learning, Deep Learning, Word Embeddings

Resumen: Presentamos diferentes métodos para análisis de sentimiento de tweets en español: SVM basado en centroide de word embeddings, CNN y LSTM. Analizamos los resultados otenidos usando el corpus de la competencia análisis de sentimiento TASS, obteniendo resultados en el estado del arte para nuestros clasificadores. Como la categoría de los neutros es la más difícil de clasificar, nos enfocamos en entender los problemas de clasificación de los neutros y analizamos la composición de esta clase en profundidad para obtener ideas sobre cómo mejorar los clasificadores.
Palabras clave: Análisis de Sentimiento, Análisis de Tweets en Español, Aprendizaje Automático, Aprendizaje Profundo, Word Embeddings

1 Introduction

Sentiment analysis is one of the most important tasks related to subjectivity analysis within Natural Language Processing. The sentiment analysis of tweets is especially interesting due to the large volume of information generated every day, the subjective nature of most messages, and the easy access to this material for analysis and processing. The existence of specific shared tasks related to this field, for several years now, shows the interest of the NLP community in working on this subject. The International Workshop on Semantic Evaluation (SemEval) includes a task on Tweets Sentiment Analysis since 2013[1]. For Spanish, the TASS workshop, organized by the SEPLN (Sociedad Española para el Procesamiento del Lenguaje Natural), focuses on this task since 2012[2].

These tasks have provided different resources available for research. The TASS workshop has generated several corpora for Sentiment Analysis in Spanish tweets, including different variants of Spanish. SemEval 2018 has also provided a corpus of Spanish tweets with polarities, but using a different set of classes than TASS.

Tweets present some special characteristics that must be taken into account: colloquial language, spelling errors, syntactic errors, abbreviations, use of symbols and URLs, lack of context, references to public events and personalities not explicitly mentioned, etc. Some of these problems can be easily overcome, but others, like syntactic errors or lack of context, have no simple solutions.

The most common approaches in recent years for tweets sentiment analysis are based on word embeddings and neural networks, to the detriment of tools for linguistic analysis, typically used to generate features for training ML algorithms, which had been applied in previous years. In many cases, information

[1] https://www.cs.york.ac.uk/semeval-2013/task2.html

[2] http://www.sepln.org/workshops/tass/2012/

ISSN 1135-5948. DOI 10.26342/2020-64-13

provided by subjective lexicons is still used.

During the TASS competitions, the measure used to rank the systems is the macro-F1 measure, which averages the F1 score for each class giving equal weight to all classes. Because of this, the participants tend to optimize their systems looking for the best macro-F1 score. However, the systems presented in the latest editions of the TASS workshop do not reach good results, thus leaving space for improvements. In particular, some papers report especially low results in neutral tweets classification. This could probably happen due to the unclear definition of these tweets, which frequently contain positive and negative fragments, resulting in a neutral global content; or due to the scant number of neutral tweets generally present in the training corpora. In any case, this low performance in the neutral class heavily penalizes the macro-F1 scores, even if the neutral tweets are not the most abundant class in the test corpus. One of the aims of this work is to try to analyze this neutral category in order to gain insights about its composition and the behavior of the classifiers around it.

In this paper we describe different approaches for Spanish tweet classification: an SVM-based classifier which uses a set of features, including word embeddings; and two deep neural network approaches: CNN and LSTM. We analyze the results obtained by each method, focusing on the improvement of neutral tweets classification.

The rest of this paper is organized as follows. Section 2 shows some background on this task and relevant related work. Section 3 describes the corpus we used and the preprocessing we made in order to work with it. Section 4 presents the linguistic resources we used to build our classifiers. Section 5 describes the three types of classifiers we used. Section 6 shows the results on the test corpus and analyzes the behavior of the classifiers, particularly over the neutral class. Finally, section 7 presents our conclusions and some future work.

2 Related Work

As in many NLP areas, in the last years most of the works on sentiment analysis have incorporated techniques based on Deep Learning and Word Embeddings, in search of improving results. In a review of Spanish Sentiment Analysis, covering works from 2012 to 2015 (Miranda and Guzmán, 2017), none of the described approaches is based on these methods, however, in recent editions of the TASS shared tasks (2017, 2018, 2019), the majority of participating systems rely on different neural network models and on the use of word embeddings (Manuel C. Díaz-Galiano, 2018; Martínez-Cámara et al., 2018; Díaz-Galiano et al., 2019; Martínez-Cámara et al., 2017). However, approaches based on classic machine learning models (like SVM), when including word embedding based features, remain competitive, reaching the top positions for some test corpora (Martínez-Cámara et al., 2018).

In TASS 2018 (task 1) three different corpora were available, each one for a different Spanish variant: Spain, Costa Rica, and Peru. The tagset used for tweets annotation included positive (`P`), negative (`N`), neutral (`NEU`) and no-opinion (`NONE`) classes. The best results were obtained by systems which used deep learning (Chiruzzo and Rosá, 2018; González, Pla, and Hurtado, 2018), SVM (Chiruzzo and Rosá, 2018), and genetic algorithms combined with SVM (Moctezuma, 2018). All of them used word embeddings for words and tweets representation. Results for monolingual experiments (using a single Spanish variant) were better than results for crosslingual experiments. As in previous TASS editions, neutral tweets are the most difficult to recognize.

SemEval 2018 (Mohammad et al., 2018) has included for the first time a dataset for Spanish tweets sentiment analysis. The corpus used in task 1.4 (ordinal classification of sentiment) is annotated with 7 values, indicating different levels of positive or negative sentiment. The best results for Spanish were obtained by systems based on deep neural networks (Convolutional Neural Networks and Recurrent Neural Networks) and SVM, based on word embeddings (Kuijper, Lenthe, and Noord, 2018; Rozental and Fleischer, 2018; Abdou, Kulmizev, and Ginés i Ametllé, 2018; González, Hurtado, and Pla, 2018). Some of them augmented the training set by translating English tweets (Kuijper, Lenthe, and Noord, 2018; Rozental and Fleischer, 2018). Other systems used subjective lexicons (Spanish lexicons and translated English lexicons).

3 Corpus

The experiments presented in this work are based on the corpora provided by different editions of the TASS sentiment analysis challenge (Martínez-Cámara et al., 2018; Martínez-Cámara et al., 2017).

We used three sets of corpora for Spanish variants spoken in different countries: Spain (ES), Costa Rica (CR) and Peru (PE), from the 2018 edition of TASS, and the general TASS training data from the 2017 edition of the competition. All the corpora are annotated with four possible polarity categories per tweet: `P`, `N`, `NEU` or `NONE`.

We joined the four sets to have a unique Spanish corpus, then divided in two subsets: training (90 %) and development (10 %). We used the test corpora distributed by TASS 2018 for evaluation. Table 1 shows the sizes of the different corpora and the number of tweets for each class.

Category	Train	Dev	Test
N	3227	361	1730
NEU	1109	123	747
NONE	2257	249	657
P	3607	400	1426
Total	10200	1133	4560

Tabla 1: Size and categories distribution for the different corpora

Each corpus was pre-processed as described in (Chiruzzo and Rosá, 2018; Rosá et al., 2017). We did not include any grammatical information, like lemma, POS-tag, morphological or syntactic information.

4 Resources

The following linguistic resources were used in our experiments:

- Subjective Lexicons: union of three subjective lexicons available for Spanish (Cruz et al., 2014; Saralegi and San Vicente, 2013; Brooke, Tofiloski, and Taboada, 2009).
- Word embeddings set: 300 dimension general purpose word embeddings set, trained by (Azzinnari and Martínez, 2016).
- Word Polarity Predictor: predictor trained using the subjective lexicons, taking a word vector as input and returning a real value for its polarity (Chiruzzo and Rosá, 2018).
- Category Markers: words that occur at least 75 % times in each category (Chiruzzo and Rosá, 2018).

5 Classifiers

This section describes the three approaches we used for classifying the polarity of Spanish tweets and gives details about the training of the methods.

5.1 SVM based approach

The SVM classifier configurations are almost the same as the ones described in (Rosá et al., 2017) and (Chiruzzo and Rosá, 2018), we used these features:

- Centroid of tweet word embeddings. (300 real values)
- Lexicon based Features:
  - Polarity of the nine (average length of tweets) more relevant words of the tweet according to the polarity predictor (those whose polarities have the highest absolute value). If the tweet has less than nine words we completed the nine values repeating the polarities of the words in the tweet. (9 real values)
  - Number of words belonging to the positive and negative lexicons. (2 natural values)
  - Number of words whose vector representations are close to the mean vector of the positive and the negative lexicons. (2 natural values)
- Number of words belonging to the lists of category markers. (4 natural values)
- Features indicating if the original tweet has repeated characters or some word written entirely in upper case. (2 boolean values)
- The thirty most relevant words from the training corpus, according to a bag of words (BoW) classifier. (30 boolean values)

This final attribute set was defined experimentally, evaluating on the development corpus. We started using just BoW attributes, obtaining better accuracy and macro-F1 as we increased the number of words (we selected the most relevant words for classifying the instances). This improvement stopped when we reached about a thousand words. We then included the centroid of the tweet word embeddings getting a better performance. The best results were obtained combining both types of attributes (BoW and the tweet centroid), increasing accuracy in 13 points and macro-F1 in 12 points. The optimal combination we found uses the centroid attributes plus the thirty most relevant words according to BoW.

The remaining attributes produced small improvements in the results as they were incorporated. While none of these attributes in particular provides a substantial improvement, the inclusion of all of them increased accuracy and macro-F1 in approximately 2 points, with respect to the results obtained using only the centroid and the thirty most relevant words. In particular, we found that the contribution of the subjective lexicon is not very relevant. Figure 1 shows different combinations of BoW and lexical features.

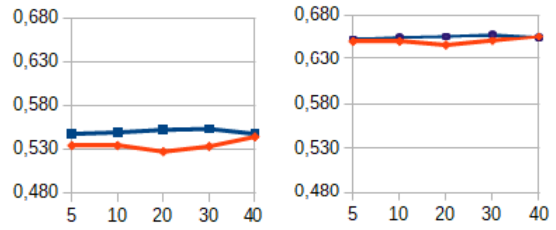


Figura 1: Macro-F1 (left) and Accuracy (right) for different combinations of BoW features, with lexical features (blue) and without lexical features (red)

In previous work, we included just five words from BoW features, in the current work, experiments on the development corpus showed that the best results are obtained using the thirty most relevant words from BoW features, most of which are words with a clear polarity, like *abrazo* (hug), *buen, buena, buenas, buenos* (different inflections of the word 'good'), *déficit* (deficit), *encanta* (love), *enhorabuena, felicidades* (variants of 'congratulations'), *feliz* (happy), *genial* (great), *gracias* (thanks), *impuestos* (taxes), *mejor* (better), *peor* (worse), *triste* (sad).

The SVM experiments were done using the *scikit-learn* toolkit (Pedregosa et al., 2011), applying the StandardScaler to the training dataset. We used the multiclass probability estimation method based on (Wu, Lin, and Weng, 2004) for training. In previous work, this method showed an improvement of 2.5 % in macro-F1 over the single class prediction.

5.2 CNN based approach

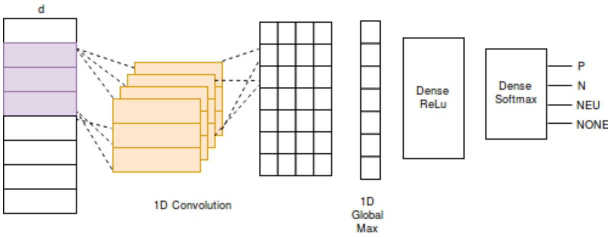


Figura 2: Overview of the convolutional neural network architecture

Convolutional Neural Networks are a class of neural networks invariant to the elements position in the input that are inspired by the cells found in the cats visual cortex. We consider two approaches using convolutional networks to perform tweets sentiment classification: 1) one branched CNN and 2) three branched CNN. The former consists in a standard CNN model, the input of the network is the sequence of word embeddings of dimension 300, corresponding to each word in the tweet, up to a maximum of 32 words. The input is fed to the 1D convolutional layer with 30 filters of dimension 256, then the output goes to a max pooling layer and a dense layer of dimension 200 with a dropout of 0,2, before going to the softmax layer for output. An overview of this architecture is shown in figure 2.

The three branched CNN is constituted by three convolutional sub-networks that consume the same input and process it independently. The three branched model considers two, three and four words of context, with 31, 30 and 29 filters of dimension 200, respectively. Then all outputs are concatenated and passed to a max pooling layer. Then, the max pooling output fed a dense layer of dimension 200 with a dropout of 0,2 before going to a softmax layer for output.

For training we keep a 70 %-30 % split for validation and use early stopping over the validation set for both networks.

5.3 LSTM based approach

Long Short-Term Memory neural networks take in consideration the whole sequence before yielding a result, they are a subclass of recurrent neural networks. In a way, what they do is calculate a sentence embedding, and then use that embedding to make a prediction. Our LSTM architecture uses the embedding for each word as input, up to a maximum of 32 words. This input is sent through a LSTM layer, for which we ignore the intermediate results, considering only the one after the whole sequence of words has been processed, we will call this output the sentence embedding. This sentence embedding of size 512 is then sent through a dense layer of size 200 with a dropout of 0.2, before getting the output through a softmax layer.

The initial experiments using this network yielded good accuracy results, but the macro-F1 measure was very low because the network did not predict any output for the class `NEU`. This class has proven to be the most difficult to learn throughout our experiments. However, we started to get better results using a different training strategy: we created two versions of the training corpus, one of them with all the tweets, and the other one taking the same number of tweets for each category (exactly the same number of tweets as the `NEU` category, which was the one with the fewest tweets). We call these sets the *full* corpus and the *balanced* corpus.

The training strategy involves training one epoch with the full corpus and one epoch with the balanced corpus, then iterate this training process until the performance over the development set stopped improving. Training the network in this fashion yields a little less accuracy but it compensates in macro-F1 measure, as it captures a lot more tweets of the `NEU` category.

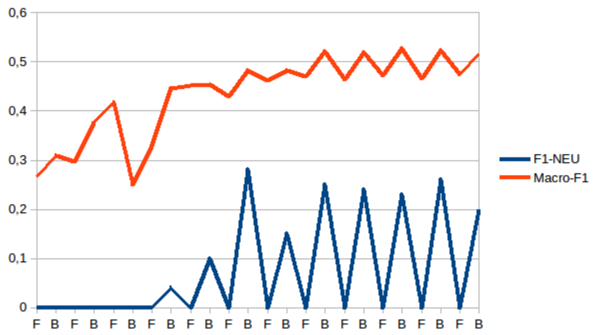


Figura 3: Macro-F1 and F1 for the `NEU` category during some iterations of the LSTM training. The iterations marked as F use the full corpus, iterations marked as B use the balanced corpus

When examining the training process more closely, we observed that after every full corpus iteration, the performance over the `NEU` category dropped to zero, while after every balanced corpus iteration the performance over that category improves. Stopping the training process after the balanced step yields better macro-F1 because it captures more `NEU` tweets. This is shown in figure 3.

Both neural network approaches (CNN and LSTM) were implemented using the *Keras* library (Chollet, 2015) and trained using the *adam* optimization algorithm (Kingma and Ba, 2014).

6 Results

In this section we present the results for each of the classifiers trained only using training data or using training and development data, then we analyze the learning curve for the classifiers and finally we discuss the case of the tweets of the neutral category in more detail.

6.1 Results for classifiers

Table 2 shows the performance of the four classifiers trained using only training data and evaluating over the development and test sets. The SVM approach clearly outperforms the other classifiers in terms of accuracy, and on the development set is also the approach with highest macro-F1. Over the test set, however, both accuracy and macro-F1 drop significantly for all classifiers. In this case, the LSTM and the three-branched CNN approaches are the ones with the best macro-F1.

	Dev		Test	
Classifier	Acc	F1	Acc	F1
SVM	65.6	55.2	**56.2**	45.4
CNN1	55.3	49.7	48.9	44.7
CNN3	59.3	52.1	55.7	46.4
LSTM	55.9	52.9	49.9	**46.6**

Tabla 2: Results for development and test corpora training only with train data

When using the whole training and development sets for training and evaluating over the test set, as shown in table 3, the ranking between classifiers is similar but the figures change. The best accuracy is still achieved by the SVM approach and is almost the same as before, while the best macro-F1 is still achieved by the LSTM, but in this case the measure is improved by two points.

As can be seen in table 4, one of the reasons the LSTM could have gotten better

Classifier	Acc	F1
SVM	**56.3**	45.3
CNN1	52.2	43.7
CNN3	51.6	47.1
LSTM	52.1	**48.7**

Tabla 3: Results for test corpora training with train and development data

results over the test set was because it could capture more tweets of the `NEU` category. This could be explained in part due to the different training strategy that focuses on giving the `NEU` tweets more weight. As we can see, the network captures more `NEU` tweets because it learned to predict a more balanced distribution. This strongly penalizes the accuracy, but is good for the macro-F1.

The three-branched CNN approach has the second best macro-F1 score, but when training with both training and development corpora its accuracy dropped respect to the other classifiers. The SVM approach is the most stable one, as its performance over the test set did not change significantly in the two models.

		Predicted			
		N	NEU	NONE	P
SVM	N	**1182**	55	258	235
Real	NEU	273	**49**	210	215
	NONE	157	14	**289**	197
	P	195	27	159	**1045**
CNN1	N	**1039**	151	244	296
Real	NEU	211	**97**	182	257
	NONE	153	50	**238**	216
	P	165	86	167	**1008**
CNN3	N	**1093**	319	222	96
Real	NEU	227	**242**	193	85
	NONE	142	131	**269**	115
	P	169	314	193	**750**
LSTM	N	**936**	489	184	121
Real	NEU	168	**305**	141	133
	NONE	97	180	**256**	124
	P	89	341	118	**878**

Tabla 4: Confusion matrix for the classifiers trained with training and development data. The LSTM captures significantly more neutral tweets

The accuracy and macro-F values we obtained are similar to the results reported in the TASS sentiment analysis challenge (Martínez-Cámara et al., 2018). We have slightly lower results than TASS monolingual experiments, and almost the same results as cross-lingual experiments. This is an expected result, since we used a cross-lingual corpus (composed of Spain, Costa Rica and Peru corpora) for all our experiments.

6.2 Learning curves

In order to see if using more annotated data would enable us to further improve our results, we experimented with varying corpus sizes, increasing the corpus size by 10 % and analyzing the behavior of the different classifiers at each step. In these experiments, we used the training and development sets together for training and we tested against the test corpus.

Figure 4 shows the macro-F1 measure for each classifier using different training sizes. The SVM is clearly improving linearly with the training set size, which indicates that using more data would keep improving this metric. The LSTM and CNN with one branch start to show better results using around 60 % of the corpus, and they also seem to keep improving given more data. The CNN with three branches seems to plateau around 80 % of the training corpus.

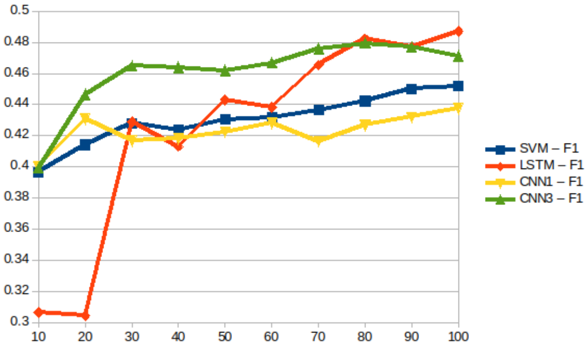


Figura 4: Macro-F1 of the classifiers trained with fractions of the corpus

6.3 Neutral tweets analysis

According to the guidelines defined for the annotation of the TASS corpus, the neutral class contains two very different subclasses: tweets without polarity, which we could call true neutrals (`NEU-NEU`), and tweets with positive parts and negative parts (`NEU-MIX`), in which none of the two polarities has a clear preponderance over the other. Two examples of these two subclases are:

- `NEU-NEU`: *@user se cosas de tiempos actuales* (@user I know things about current times)

- NEU-MIX: *feo es tener clases un feriado, por suerte yo no tengo* (it's ugly to have classes on a holiday, fortunately I don't)

While the overall results for neutral tweets are quite different for the different classifiers (as already stated, the LSTM-based system obtains a noticeably better result with this class than the other two), the three models behave in a similar way regarding the two subclasses of neutral tweets (CNN with one branch is not analyzed in this section).

Class	Real	SVM	CNN3	LSTM
NEU	747	7 %	30 %	41 %
NEU-NEU	276	4 %	29 %	31 %
NEU-MIX	471	8 %	31 %	48 %

Tabla 5: Percentage of correct predictions of Neutral tweets and subclasses NEU-NEU and NEU-MIX (training only with the train corpus)

	SVM	CNN3	LSTM
NEU-NEU vs NONE	121	101	85
NEU-NEU vs N	93	41	66
NEU-NEU vs P	50	54	40
NEU-MIX vs NONE	88	87	44
NEU-MIX vs N	191	113	124
NEU-MIX vs P	151	126	79

Tabla 6: NEU-NEU and NEU-MIX confusion with other classes

In all cases, the systems are better at recognizing tweets of the NEU-MIX category than those of the NEU-NEU category (see table 5). In addition, NEU-NEU tweets are confused by the three classifiers with the NONE class more frequently than with the N and P classes (see table 6). From a human point of view it is also difficult to distinguish the NEU-NEU class, which indicates that the tweet is subjective but with no polarity, from the NONE class, which indicates that the tweet transmits non-subjective information (for example, *Último recibo que pagaré será el de Setiembre. / The last receipt that I will pay will be the one of September*). On the other hand, the NEU-MIX tweets are often confused with the N and P classes. This behavior seems to be due to the fact that the NEU-MIX tweets contain portions with marked polarity (both positive and negative), while the NEU-NEU tweets should not contain a clear polarity at all.

7 Conclusion

We presented three approaches for classifying the sentiment of Spanish tweets. The approaches we used are: SVM using word embedding centroids and manually crafted features, one and three branched CNNs using word embeddings as input, and LSTM using word embeddings, trained with focus on improving the recognition of neutral tweets. None of the classifiers was a clear winner in our experiments. However, we found that the training method used for the LSTM significantly improved its macro-F1 measure by improving the detection of neutral tweets. In all cases, the use of word embeddings was key to improve the performance of the methods.

Analyzing the learning curves of the classifiers, we concluded that most of them would still keep improving if there was more data available. However, it is clear that the bottleneck of the macro-F1 performance is still the neutral class.

We separated neutral tweets in two classes, NEU-MIX and NEU-NEU, and we analyzed classifiers mistakes on each class. We found that NEU-MIX tweets are usually confused with negative and positive classes, while NEU-NEU tweets are confused with the class NONE. We propose as future work to improve neutral tweets classification by segmenting the input into chunks with homogeneous sentiment polarity and feeding this chunks as units to the classifiers.

References

Abdou, M., A. Kulmizev, and J. Ginés i Ametllé. 2018. Affecthor at semeval-2018 task 1: A cross-linguistic approach to sentiment intensity quantification in tweets. In *12th International Workshop on Semantic Evaluation*, pages 210–217. ACL.

Azzinnari, A. and A. Martínez. 2016. Representación de Palabras en Espacios de Vectores. Proyecto de grado, Universidad de la República, Uruguay.

Brooke, J., M. Tofiloski, and M. Taboada. 2009. Cross-linguistic sentiment analysis: From english to spanish. In *RANLP*, pages 50–54.

Chiruzzo, L. and A. Rosá. 2018. RETUYT-InCo at TASS 2018: Sentiment Analysis in Spanish Variants using Neural Networks and SVM. In *TASS 2018*.

Chollet, F. 2015. Keras. https://github.com/fchollet/keras.

Cruz, F. L., J. A. Troyano, B. Pontes, and F. J. Ortega. 2014. Building layered, multilingual sentiment lexicons at synset and lemma levels. *Expert Systems with Applications*, 41(13):5984–5994.

Díaz-Galiano, M. C., M. García-Vega, E. Casasola, L. Chiruzzo, M. A. García-Cumbreras, E. Martínez-Cámara, D. Moctezuma, A. Montejo Ráez, M. A. Sobrevilla Cabezudo, E. Tellez, M. Graff, and S. Miranda. 2019. Overview of TASS 2019: One more further for the global spanish sentiment analysis corpus. In *IberLEF 2019*, Bilbao, Spain, September.

González, J., F. Pla, and L. Hurtado. 2018. ELiRF-UPV at TASS 2018: Sentiment Analysis in Twitter based on Deep Learning. In *TASS 2018*.

González, J.-Á., L.-F. Hurtado, and F. Pla. 2018. Elirf-upv at semeval-2018 tasks 1 and 3: Affect and irony detection in tweets. In *12th International Workshop on Semantic Evaluation*, pages 565–569. ACL.

Kingma, D. P. and J. Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Kuijper, M., M. Lenthe, and R. Noord. 2018. Ug18 at semeval-2018 task 1: Generating additional training data for predicting emotion intensity in spanish. In *12th International Workshop on Semantic Evaluation*, pages 279–285. ACL.

Manuel C. Díaz-Galiano, E. M.-C. y. M. Á. G. C. y. M. G. V. y. J. V. R. 2018. The democratization of deep learning in tass 2017. *Procesamiento del Lenguaje Natural*, 60(0):37–44.

Martínez-Cámara, E., Y. Almeida-Cruz, M. C. Díaz-Galiano, S. Estévez-Velarde, M. A. García-Cumbreras, M. García-Vega, Y. Gutiérrez, A. Montejo Ráez, A. Montoyo, R. Muñoz, A. Piad-Morffis, and V.-R. Julio. 2018. Overview of TASS 2018: Opinions, health and emotions. In *TASS 2018*, volume 2172 of *CEUR Workshop Proceedings*, Sevilla, Spain, September. CEUR-WS.

Martínez-Cámara, E., M. C. Díaz-Galiano, M. A. García-Cumbreras, M. García-Vega, and J. Villena-Román. 2017. Overview of tass 2017. In *TASS 2017*, volume 1896 of *CEUR Workshop Proceedings*, Murcia, Spain, September. CEUR-WS.

Miranda, C. H. and J. Guzmán. 2017. A Review of Sentiment Analysis in Spanish. *Tecciencia*, 12:35 – 48, 06.

Moctezuma, D., O.-B. J. T.-E. M.-J. S. G. M. 2018. INGEOTEC solution for Task 1 in TASS'18 competition. In *TASS 2018*.

Mohammad, S., F. Bravo-Marquez, M. Salameh, and S. Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *12th International Workshop on Semantic Evaluation*, pages 1–17.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Rosá, A., L. Chiruzzo, M. Etcheverry, and S. Castro. 2017. RETUYT en TASS 2017: Análisis de Sentimientos de Tweets en Español utilizando SVM y CNN. In *TASS 2017*.

Rozental, A. and D. Fleischer. 2018. Amobee at semeval-2018 task 1: Gru neural network with a cnn attention mechanism for sentiment classification. In *12th International Workshop on Semantic Evaluation*, pages 218–225. ACL.

Saralegi, X. and I. San Vicente. 2013. Elhuyar at tass 2013. *XXIX Congreso de la Sociedad Espaola de Procesamiento de lenguaje natural, Workshop on Sentiment Analysis at SEPLN (TASS2013)*, pages 143–150.

Wu, T.-F., C.-J. Lin, and R. C. Weng. 2004. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5(Aug):975–1005.

Tesis

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 119-122 recibido 18-10-2019 revisado 12-01-2020 aceptado 15-01-2020

Adverse Drug Reaction extraction on Electronic Health Records written in Spanish

Extracción de Reacciones Adversas a Medicamentos en Historias Clínicas Electrónicas escritas en español

Sara Santiso González
IXA group, University of the Basque Country (UPV/EHU)
Manuel Lardizabal 1, 20018, Donostia
sara.santiso@ehu.eus

Abstract: PhD thesis on Language Analysis and Processing written by Sara Santiso González at the University of the Basque Country (UPV/EHU) under the supervision of Dr. Arantza Casillas (Department of Electricity and Electronic) and Dr. Alicia Pérez (Department of Computer Languages and Systems). The thesis defense was held on June 13, 2019 and the members of the commission were Dr. Raquel Martinez (President, National Distance Education University (UNED)), Dr. Arantza Díaz de Ilarraza (Secretary, University of the Basque Country (UPV/EHU)) and Dr. Lluis Padró (Vocal, Technical University of Catalonia (UPC)). The thesis obtained excellent grade with Cum Laude mention.

Keywords: Adverse Drug Reactions, Electronic Health Records, Text mining, Supervised machine learning

Resumen: Tesis doctoral en Análisis y Procesamiento del Lenguaje defendida por Sara Santiso González en la Universidad del País Vasco (UPV/EHU) y realizada bajo la dirección de las doctoras Arantza Casillas Rubio (Departamento de Electricidad y Electrónica) y Alicia Pérez Ramírez (Departamento de Lenguajes y Sistemas Informáticos). La defensa de la tesis tuvo lugar el 13 de junio de 2019 ante el tribunal formado por los doctores Raquel Martinez (Presidenta, Universidad Nacional de Educación a Distancia (UNED)), Arantza Díaz de Ilarraza (Secretaria, Universidad del País Vasco (UPV/EHU)) y Lluis Padró (Vocal, Universidad Politécnica de Cataluña (UPC)). La tesis obtuvo la calificación de sobresaliente con mención Cum Laude.

Palabras clave: Reacciones Adversas a Medicamentos, Historias Clínicas Electrónicas, Minería de textos, Apredizaje automático supervisado

1 Introduction

This thesis was developed on the IXA group of the University of the Basque Country (UPV/EHU) and is related with the extraction of Adverse Drug Reactions (ADRs).

An ADR is defined by the World Health Organization (WHO) as 'a response to a medicine which is noxious and unintended, and which occurs at doses normally used in man'. The WHO informed about the importance of reporting ADRs to understand and treat the diseases caused by drugs and, as a result, improve the patients care. However, ADRs are still heavily under-reported, which makes their prevention difficult. This was the **motivation** to automatically extract ADRs on Electronic Health Records (EHRs). Given that information stored digitally by the hospitals is growing, Natural Language Processing (NLP) techniques can be used to create a system that helps the doctors to analyze the ADRs of the patients in a given EHR, facilitating the decision making process and alleviating the work-load. As a consequence, the patients' health could improve and the pharmaco-surveillance service would be informed about the detected ADRs. The ADR extraction was defined as a relation extraction task. That is, the aim is to detect ADR relations between the entities (drugs and diseases) recognized in a given text. For the ADR extraction developed in this work, we

ISSN 1135-5948. DOI 10.26342/2020-64-14

distinguished two steps that were developed using a pipeline approach:

1. Medical Entity Recognition (MER) to find "drug" entities and "disease" entities. The "drug" entity encompasses either a brand name, a substance or an active ingredient and the "disease" entity encompasses either a disease, a sign or a symptom.
2. ADR detection to discover the relations between "drug" entities and "disease" entities that correspond to ADRs. The "drug" entity would be the causative agent and the "disease" entity would be the caused adverse reaction.

In the ADR extraction process, we had to overcome some challenges that make this supervised classification task difficult. On the one hand, the ADRs are minority relations because generally the drug and the disease are either unrelated or related as treatment and, thus, the ADRs are rare cases. On the other hand, the EHRs show multiple lexical variations. In addition, our EHRs are written in Spanish whereas the majority of biomedical NLP research has been done in English.

In this way, we tackled some drawbacks present in state-of-the-art works: the class imbalance, the lexical variability and the few resources and tools to apply NLP in the medical domain for Spanish and other languages different to English.

The main **objective** of this work is the creation of a model able to detect automatically ADRs in EHRs written in Spanish. This, in turn, encompasses the sub-objectives stated below:

- Detect ADRs by discovering relations between the causative drug and the caused diseases.

  The aim is to detect drug-disease pairs related as ADRs and not only the disease caused by the drug. Indicating explicitly the entities involved in an ADR can result more useful for their study.

- Discover approaches to overcome the class imbalance.

  Given that ADRs are rare events, it is frequent to find the class imbalance problem. Machine learning algorithms tend to expect balanced class distributions and learning the minority class is difficult for them. For this reason, our intention is to explore different techniques that could help to tackle this issue improving the ADR detection or find approaches that could be robust against imbalanced distributions of the class.

- Discover robust representations to cope with the lexical variability and the data sparsity.

  This is a challenge goal due to two factors. First, the EHRs are written during consultation time and each doctor uses different terms or expressions, producing lexical variations. Second, due to confidentiality issues, there is a lack of available EHRs. Then, our intention is to explore different representations to make the most of the annotated corpus.

2 Thesis Overview

This thesis was organized in eight chapters.

Chapter 1 explains the motivation to develop the ADR extraction and the framework. It also presents the objectives to achieve together with the main research question to address.

Chapter 2 makes a review of the works related with the ADR extraction task. It focuses on the definition of ADR extraction, the techniques and features employed for the ADR classification, the corpora and the evaluation schemes used for ADR extraction.

Chapter 3 presents the corpora employed in this work (IxaMed-GS, IxaMed-CH, IxaMed-E). Furthermore, it describes the schemes and metrics employed for the evaluation of our systems.

Chapter 4 describes the features employed to create the symbolic characterizations of the ADR events, our first approach. It presents the Random Forest classifier used for ADR detection of intra-sentence as well as inter-sentence ADRs. It also explains the approaches explored to tackle the class imbalance (Santiso et al., 2014; Santiso et al., 2016; Casillas et al., 2016b; Casillas et al., 2016a; Santiso, Casillas, and Pérez, 2019). With this approach, the best results are a precision of 34.0, a recall of 59.3 and an f-measure of 43.2.

Chapter 5 explains the dense characterizations created from embeddings that were used together with the Random Forest classifier overcoming the class imbalance, our second approach. Moreover, it proposes different smoothing techniques that were applied to the dense representations in order to improve the proximity between semantically related words (Santiso, Pérez, and Casillas, 2019b). With this approach, the best results are a precision of 47.4, a recall of 66.7 and an f-measure of 55.4.

Chapter 6 explains the neural networks used for ADR detection, Joint AB-LSTM networks, as our third approach. It includes the core-features employed to infer the dense representations. It also presents the techniques explored to overcome the class imbalance suited for neural networks (Santiso, Pérez, and Casillas, 2019a). With this approach, the best results are a precision of 72.4, a recall of 71.4 and an f-measure of 71.9.

Chapter 7 discusses the results obtained with the best performing approach, using slightly different corpora and incorporating the automatic detection of medical entities. Until now, we have just focused on the ADR detection step and we have tried different representations and classifiers. The best results are obtained with the higher corpus, yielding a precision of 74.4, a recall of 76.0 and an f-measure of 75.2.

Chapter 8 gives the final conclusions, which include the response to the research questions and the main contributions. It explains the future lines of work regarding the ADR extraction. It also shows the publications related to this work.

In addition, it includes three appendices.

Appendix A explains the two approaches explored to detect negated entities automatically. These negated entities are used to discard negative ADR candidates (Santiso et al., 2017; Santiso et al., 2019).

Appendix B briefly explains some experiments developed to detect medical entities automatically. These entities are those used to observe the influence of MER step on ADR detection.

Appendix C gives detailed results of the experiments developed in Chapter 5 for ADR detection using dense representations and the Random Forest classifier.

3 Contributions

The main **contribution** of this work is that the ADR extraction was developed using EHRs written in Spanish. To the best of our knowledge, for ADR extraction in texts written in Spanish, we are the first employing EHRs. Other contributions derived from the tasks carried out during this work are:

- Combination of approaches to tackle the high class imbalance.

  We made a step ahead in the development of NLP methods that deal with ADR extraction defined as relation extraction task. As a first approach we tackled both inter- and intra-sentence ADR extraction, even though the mainstream in the related works just focused on intra-sentence relations. In this context, inference algorithms should be suited to cope with the challenge of an extremely high class imbalance. Although the imbalance problem diminishes considerably in intra-sentence scenarios, we explored classical approaches to tackle the class imbalance (sampling, cost-sensitive learning, ensemble learning, one-class classification) in the context of inter- and intra-sentence ADR extraction. We observed that the combination of them, precisely sampling and cost-sensitive learning, was beneficial in our framework.

  Besides, in an attempt to discard non-ADR instances and alleviate the class imbalance, we also tried negation detection. We developed two ways of detecting negated medical entities in EHRs: an adaptation of the NegEx tool and a Conditional Random Fields algorithm using dense characterizations. We corroborated, however, that class imbalance can be tackled in intra-sentence ADR extraction, while there is room for improvement in inter-sentence relation extraction.

- Mechanisms to deal with lexical variability.

  NLP in the medical domain dealing with EHRs has, among others, the challenge of high lexical variability (large specialized vocabularies, non-standard abbreviations, misspellings, etc.) and lack of available corpora. Quantitatively, there is a reflect of the lexical variability in the remarkable ratio of Out-Of-Vocabulary elements. To cope with this issue it results crucial to propose not only competitive inference algorithms but also robust characterizations of the instances.

Throughout this work we analyzed two classification techniques (Random Forest, Joint AB-LSTM) and two representations (symbolic, dense). We experimentally corroborated that context-aware embeddings (dense representations created taking into account the embeddings of the context-words) are useful to preserve the lexical nuances in this domain. In addition, to alleviate the influence that the lack of training samples might have in the quality of the inferred dense representations, we proposed the use of smoothing techniques. Smoothing helps to avoid superficial variations and, hence, makes different (but close) points in the space to be equivalent.

Moreover, we observed that dense spaces of lemmas also helped to tackle the lexical variability. In fact, lemmatization was particularly effective in the neural networks used for ADR extraction.

- Tolerance to external noise.

  We exposed the ADR extraction system to two types of noise. On the one hand, we assessed the impact of corpora from slightly different sources (different hospitals with different services or specializations). On the other hand, we analyzed the influence of miss-recognized medical entities into the ADR detection step leading to a fully automatic ADR extraction system. We corroborated that the Joint AB-LSTM is able to cope with these types of noise although, naturally, there is a small decrease in its performance due to the missed entities involved in the ADR pairs.

Acknowledgements

The author would like to thank the staff of the Pharmacy and Pharmacovigilance services of the Galdakao-Usansolo and Basurto hospitals. This work was partially funded by the Spanish Ministry of Science and Innovation (PROSAMED: TIN2016-77820-C3-1-R) and the Basque Government (BERBAOLA: KK-2017/00043, Predoctoral Grant: PRE 2018 2 0265).

References

Casillas, A., A. Díaz de Ilarraza, K. Fernandez, K. Gojenola, M. Oronoz, A. Pérez, and S. Santiso. 2016a. IXAmed-IE: Online medical entity identification and ADR event extraction in Spanish. In *2016 IEEE International Conference on Bioinformatics and Biomedicine*, pages 846–849.

Casillas, A., A. Pérez, M. Oronoz, K. Gojenola, and S. Santiso. 2016b. Learning to extract adverse drug reaction events from electronic health records in Spanish. *Expert Systems with Applications*, 61:235–245.

Santiso, S., A. Casillas, and A. Pérez. 2019. The class imbalance problem detecting adverse drug reactions in electronic health records. *Health Informatics Journal*, 25(4):1768–1778.

Santiso, S., A. Casillas, A. Pérez, and M. Oronoz. 2017. Medical entity recognition and negation extraction: Assessment of NegEx on health records in Spanish. In *International Work-Conference on Bioinformatics and Biomedical Engineering*, pages 177–188.

Santiso, S., A. Casillas, A. Pérez, and M. Oronoz. 2019. Word embeddings for negation detection in health records written in Spanish. *Soft Computing*, 23(21):10969–10975.

Santiso, S., A. Casillas, A. Pérez, M. Oronoz, and K. Gojenola. 2014. Adverse drug event prediction combining shallow analysis and machine learning. In *Proceedings of the 5th International Workshop on Health Text Mining and Information Analysis (Louhi)*, pages 85–89.

Santiso, S., A. Casillas, A. Pérez, M. Oronoz, and K. Gojenola. 2016. Document-level adverse drug reaction event extraction on electronic health records in Spanish. *Procesamiento del Lenguaje Natural*, 56(0):49–56.

Santiso, S., A. Pérez, and A. Casillas. 2019a. Exploring joint ab-lstm with embedded lemmas for adverse drug reaction discovery. *IEEE Journal of Biomedical and Health Informatics*, 23(5):2148–2155.

Santiso, S., A. Pérez, and A. Casillas. 2019b. Smoothing dense spaces for improved relation extraction between drugs and adverse reactions. *International journal of medical informatics*, 128:39–45.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 123-126 recibido 04-11-2019 revisado 11-01-2020 aceptado 12-01-2020

Identification and translation of verb+noun Multiword Expressions: A Spanish-Basque study

Identificación y traducción de Expresiones Multipalabra de tipo verbo+sustantivo: análisis de castellano-euskera

Uxoa Inurrieta
Ixa NLP group, University of the Basque Country (UPV/EHU)
usoa.inurrieta@ehu.eus

Abstract: This is a summary of the PhD thesis written by Uxoa Iñurrieta under the supervision of Dr. Gorka Labaka and Dr. Itziar Aduriz. Full title of the PhD thesis in Basque: *Izena+aditza Unitate Fraseologikoak gaztelaniatik euskarara: azterketa eta tratamendu konputazionala.* The defense was held in San Sebastian on November 29, 2019. The doctoral committee was integrated by Ricardo Etxepare (Centre National de la Recherche Scientifique), Margarita Alonso (Universidad de Coruña) and Miren Azkarate (University of the Basque Country).
Keywords: Multiword Expressions, Phraseology, Identification, Machine Translation, Basque

Resumen: Este es un resumen de la tesis doctoral escrita por Uxoa Iñurrieta bajo la supervisión del Dr. Gorka Labaka y la Dra. Itziar Aduriz. Título completo de la tesis en euskera: *Izena+aditza Unitate Fraseologikoak gaztelaniatik euskarara: azterketa eta tratamendu konputazionala.* La defensa de la tesis se celebró en Donostia-San Sebastián el 29 de Noviembre de 2019, ante el tribunal formado por Ricardo Etxepare (Centre National de la Recherche Scientifique), Margarita Alonso (Universidad de Coruña) y Miren Azkarate (UPV/EHU).
Palabras clave: Expresiones Multipalabra, Fraseología, Identificación, Traducción Automática, Euskera

1 Motivation

Multiword Expressions (MWEs) are combinations of words which exhibit some kind of lexical, morphosyntactic, semantic, pragmatic or statistical idiosyncrasy (Baldwin and Kim, 2010). Due to their idiosyncratic nature, MWEs pose important challenges to Natural Language Processing (NLP), and sophisticated strategies are needed in order to process them correctly (Constant et al., 2017). Two main types of word combinations are comprised in the category of MWEs: idioms (example 1) and collocations (example 2), the latter including light verb constructions (example 3). All of them are considered in this work.

(1) *She always ends up* ***spilling the beans****.* (lit. revealing the secret)

(2) *All students* ***passed*** *the* ***exam****.*

(3) *She is* ***giving*** *a* ***lecture*** *this afternoon.*

In this PhD, two tasks concerning MWE processing are addressed: on the one hand, their identification in corpora, and on the other hand, their processing within Machine Translation (MT).

Automatic identification of MWEs involves finding occurrences of previously known MWEs (Ramisch et al., 2018). For instance, considering the MWE *make conclusions*, occurrences would need to be identified in examples 4a–c, but not in 4d. Likewise, for the MWE *pull somebody's leg*, an identification system would need to distinguish MWE occurrences like the one in example 5a from non-MWEs like the one in example 5b. In order to do so, it must be taken into account that many MWEs, especially verbal ones, tend to be morphosyntactically variable, discontiguous, and ambiguous depending on the context.

(4) a. *They* ***made*** *a* ***conclusion****.*
b. *They* ***made*** *some simple but*

ISSN 1135-5948. DOI 10.26342/2020-64-15

interesting ***conclusions*.**

c. *The* ***conclusions*** *they* ***make*** *are always interesting.*

d. *They will* <u>*make*</u> *progress and will come to a* <u>*conclusion*</u>*.*

(5) a. *She is not serious. She is just* ***pulling your leg*.**

b. *Grab your knee,* <u>*pulling your leg*</u> *toward your chest.*

Concerning Machine Translation, apart from identifying MWEs in the source text, an appropriate equivalent must be given to them in the target language. This poses additional challenges, since word co-occurrence is often arbitrary (example 6) and some MWEs are non-compositional (example 7), meaning that word-for-word translations are incorrect in many cases.

(6) EN: ***pay attention***
ES: ***prestar atención***
(lit. lend attention)
EU: ***arreta jarri***
(lit. put attention)

(7) EN: ***pull*** *somebody's* ***leg***
ES: ***tomar el pelo*** *a alguien*
(lit. take sb's hair)
EU: *norbaiti* ***adarra jo***
(lit. play the horn to sb)

The main assumption behind the work in this PhD is that specific linguistic information (mainly lexical and morphosyntactic data) is helpful for MWE identification and translation of MWEs within MT. An in-depth linguistic analysis and several experiments were undertaken to verify this, which are outlined in Section 2. Then, in Section 3, the hypotheses covered in the PhD are listed and the main contributions are briefly described.

2 General outline of the dissertation

The contents of the dissertation can be divided into five parts. A brief overview will now be given, and the papers linked to each of the parts will be specified when available.

- Preanalysis (Iñurrieta et al., 2018a). A study based on a Spanish-Basque bilingual dictionary was undertaken, where verb+noun entries and their translations were analysed along lexical and morphosyntactic dimensions. Phraseological differences and similarities of both languages were made evident, as well as the prevalence of non-word-for-word translations concerning verb+noun MWEs.
- Manual analysis (Iñurrieta et al., 2016; Iñurrieta et al., 2017). An in-depth manual analysis of a set of Spanish MWEs and their Basque translations was carried out. Lexical and morphosyntactic data were considered, which were then employed in two experiments to see how they affected identification and MT. Results were positive in both cases.
- (Semi)automatic analysis. Considering the good results of the experiments using manual data, but taking into account that fully manual studies are limited and expensive, an improved analysis method was proposed to extract linguistic data about MWEs and their translations from corpora. Part of the information was then manually corrected to evaluate the different steps of the method. Finally, two more experiments were undertaken, where the positive impact of the gathered data was confirmed for identification and, to a lesser extent, for MT.
- The *Konbitzul* database (Iñurrieta et al., 2018b). All the linguistic data gathered from the previous studies were collected in a publicly accessible database: *Konbitzul*. Information can be either queried online or downloaded to be used for NLP purposes.
- PARSEME-related work (Iñurrieta et al., 2018; Savary et al., 2019). Verbal MWEs were annotated in a Basque corpus, which was integrated into the PARSEME multilingual corpus. A study of literal occurrences of MWEs was then undertaken based on the previously annotated corpus.

3 Hypotheses and contributions

Several hypotheses were proposed and validated through the work outlined in the previous section. These hypotheses are listed below (Section 3.1), and the main contributions of the research undertaken are then summarised (Section 3.2).

3.1 Hypotheses

Six hypotheses were tested and confirmed in this PhD, four of which were mostly related to linguistic aspects of MWEs, and the remaining two, to MWE processing. All six hypotheses are listed below. A summary of the main evidence found to support them can be found in the English version of the PhD thesis (Inurrieta, 2019, pp. 12–14).

[**H1**] Many MWEs are not translated word-for-word from one language to another.

[**H2**] Verbal MWEs tend to be rather flexible concerning morphosyntax, although not completely, since they also have some restrictions.

[**H3**] Compared to many other languages which have been analysed from a phraseological perspective, light verb constructions are especially frequent in Basque.

[**H4**] Although many word combinations can be idiomatic or literal depending on the context, very few of them are actually used literally in real texts.

[**H5**] Detailed morphosyntactic information is helpful for MWE identification.

[**H6**] MWE-specific linguistic information is beneficial for MT.

3.2 Contributions

Apart from confirming the hypotheses listed above, a number of contributions were made through the work in this PhD. The main ones are listed below.

[C1] Comprehensive NLP-applicable study of verb+noun MWEs in Spanish and Basque. Although there exist other studies on verb+noun MWEs in both languages, the one in this PhD differs from them in two main aspects. On the one hand, because it is NLP-oriented, unlike most of the phraseological analyses carried out for Spanish and Basque. On the other hand, because most of the data obtained from it is quantified, which is helpful to see the extent of the MWE-specific features under study. Furthermore, as will be shown in contribution 6, all data were made publicly available.

[C2] Analysis of the translation of verb+noun MWEs between Spanish and Basque. Almost no research has been undertaken about phraseology in Spanish-Basque translation, and this work brings a contribution into the field. The verb+noun entries and translations in the *Elhuyar* dictionary were firstly analysed, and translations were automatically extracted from parallel corpora for further MWEs from other sources. In both analyses, lexical and morphosyntactic features were examined, to see how these change when MWEs are translated from one language to the other.

[C3] Proposal or application of methodologies which are adaptable to other languages. The idea of replicability and reusability of our methods was present throughout the whole work. Firstly, in order to test whether the proposed analysis was applicable to other languages, the manual study of Spanish verb+noun MWEs was undertaken also in English. The output data was then used for an MWE identification experiment, where results were even better than the Spanish ones. Secondly, part of the method proposed to automatise the analysis of MWEs was reused on Basque corpora to gather translation-oriented information. Only a few modifications needed to be done, which means that it is easily adaptable. Thirdly, the PARSEME universal guidelines were followed to annotate verbal MWEs in a Basque corpus, just like in 19 other languages. And finally, the study of literal occurrences of MWEs was done in five languages of different phylogenetic families.

[C4] Improvement of the identification of verb+noun MWEs. The identification method proposed in this PhD outperforms all results in the Spanish part of the PARSEME shared task edition 1.1, with an F score of 0.51, which is 13 points higher than the best result in the Spanish task. Besides, it was made evident precisely what morphosyntactic features are helpful for identification.

[C5] Integration of MWE-specific linguistic data into MT. Lexical and morphosyntactic information specific to a set of verb+noun MWEs was added to the *Matxin* rule-based MT system, and results were better than the basic system both according to a manual evaluation (improvement of 62-65%) and according to statistical measures (increase of 2.25% in BLEU).

[C6] Creation of a database collecting all MWEs and translations covered in this work, along with NLP-applicable

linguistic data. The *Konbitzul* database is publicly accessible online[1]. Its interface enables users to make queries according to several criteria and filters, and all NLP-applicable information can be fully downloaded. In all, 1,927 Spanish MWEs (along with 4,043 translations) and 2,074 Basque MWEs (along with 3,022 translations) are collected in it, out of which 894 Spanish MWEs and their corresponding translations contain NLP-applicable information.

[C7] **Annotation of Spanish and (especially) Basque corpora from a phraseological perspective.** The PARSEME multilingual corpus comprises texts of 20 different languages, including Spanish and Basque. Its annotation was carried out in two phases, and we contributed to both of them: in the first edition, as part of the Spanish annotation team; in the second one, by creating the Basque corpus, which consists of 11,158 sentences (157,807 words) and 3,823 MWE annotations. Then, literal occurrences were also studied and annotated on five of the languages in the PARSEME corpus, including Basque. Both the original PARSEME corpus and the one including annotations about literal occurrences are available online[2].

Acknowledgements

The Spanish Ministry of Economy and Competitiveness, who awarded Uxoa Iñurrieta a predoctoral fellowship (BES-2013-066372) to conduct research within the SKATeR project (TIN2012-38584-C06-02).

References

Baldwin, T. and S. N. Kim. 2010. Multiword Expressions. *Handbook of Natural Language Processing*, 2:267–292.

Constant, M., G. Eryiğit, J. Monti, L. Van Der Plas, C. Ramisch, M. Rosner, and A. Todirascu. 2017. Multiword Expression processing: a survey. *Computational Linguistics*, 43(4):837–892.

Inurrieta, U. 2019. *Verb+Noun Multiword Expressions: A linguistic analysis for identification and translation.* Ph.D. thesis, University of the Basque Country (UPV/EHU).

Iñurrieta, U., I. Aduriz, A. Díaz de Ilarraza, G. Labaka, and K. Sarasola. 2017. Rule-based translation of Spanish verb-noun combinations into Basque. In *Proceedings of the 13th Workshop on Multiword Expressions*, pages 149–154. Valencia, Spain.

Iñurrieta, U., I. Aduriz, A. Díaz de Ilarraza, G. Labaka, and K. Sarasola. 2018a. Analysing linguistic information about word combinations for a Spanish-Basque rule-based Machine Translation system. In *Multiword Units in Machine Translation and Translation Technologies.* John Benjamins Publishing C., pages 41–60.

Iñurrieta, U., I. Aduriz, A. Díaz de Ilarraza, G. Labaka, and K. Sarasola. 2018b. Konbitzul: an MWE-specific database for Spanish-Basque. In *Proceedings of the 11th Language Resources and Evaluation Conference (LREC2018)*, pages 2500–2504. Miyazaki, Japan.

Iñurrieta, U., I. Aduriz, A. Díaz de Ilarraza, G. Labaka, K. Sarasola, and J. Carroll. 2016. Using linguistic data for English and Spanish verb-noun combination identification. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING2016): Technical Papers*, pages 857–867. Osaka, Japan.

Iñurrieta, U., I. Aduriz, A. Estarrona, I. Gonzalez-Dios, A. Gurrutxaga, R. Urizar, and I. Alegria. 2018. Verbal Multiword Expressions in Basque corpora. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions*, pages 86–95. Santa Fe, New Mexico, USA.

Ramisch, C., S. Cordeiro, A. Savary, et al. 2018. Edition 1.1 of the parseme shared task on automatic identification of verbal multiword expressions. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions*, pages 222–240. Santa Fe, New Mexico, USA.

Savary, A., S. R. Cordeiro, T. Lichte, C. Ramisch, U. Iñurrieta, and V. Giouli. 2019. Literal occurrences of multiword expressions: rare birds that cause a stir. *The Prague Bulletin of Mathematical Linguistics*, 112:5–54.

[1] http://ixa.eus/node/4484

[2] PARSEME multilingual corpus of verbal MWEs: http://hdl.handle.net/11372/LRT-2842. Corpus of literal occurrences: http://hdl.handle.net/11372/LRT-2966.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 127-130 recibido 30-11-2019 revisado 12-01-2020 aceptado 16-01-2020

A comprehensive analysis of the parameters in the creation and comparison of feature vectors in distributional semantic models for multiple languages

Análisis exhaustivo respecto a varios idiomas de los parámetros usados durante la creación y comparación de vectores de propiedades de modelos semánticos distribucionales

András Dobó
Institute of Informatics, University of Szeged
2 Árpád tér, Szeged, 6720 Hungary
dobo@inf.u-szeged.hu

Abstract: PhD thesis written by András Dobó under the supervision of Prof. Dr. János Csirik (University of Szeged). The thesis was defended in Szeged (Hungary) on the 15th of November, 2019. The doctoral committee comprised of Prof. Dr. Márk Jelasity (University of Szeged), Prof. Dr. András Kornai (Budapest University of Technology and Economics), Prof. Dr. Reinhard Köhler (University of Trier), Dr. Rudolf Ferenc (University of Szeged) and Dr. Zsolt Gazdag (University of Szeged). The thesis obtained the grade of *Summa Cum Laude*.

Keywords: distributional semantic models; semantic similarity and relatedness; best combination of parameter settings; English, Spanish and Hungarian

Resumen: Tesis doctoral elaborada por András Dobó con la supervisión del Prof. Dr. János Csirik (Universidad de Szeged). La defensa de la tesis tuvo lugar en Szeged (Hungría), el 15 de noviembre de 2019. Los miembros del comité de doctorado fueron Prof. Dr. Márk Jelasity (Universidad de Szeged), Prof. Dr. András Kornai (Universidad de Tecnología y Economía de Budapest), Prof. Dr. Reinhard Köhler (Universidad de Tréveris), Dr. Rudolf Ferenc (Universidad de Szeged) y Dr. Zsolt Gazdag (Universidad de Szeged). La tesis fue evaluada con la calificación de *Summa Cum Laude*.

Palabras clave: modelos de semántica distribucional; similitud y relación semántica; combinación óptima de configuraciones de parámetros; inglés, español y húngaro

1 Introduction

For many natural language processing problems, including noun compound interpretation (Dobó and Pulman, 2011), it is crucial to determine the semantic similarity or relatedness of words. While relatedness considers a wide range of relations (including similarity), similarity only examines how much the concepts denoted by the words are truly alike.

1.1 Motivation

Most semantic models calculate the similarity or relatedness of words using distributional data extracted from large corpora. These models can be collectively called as distributional semantic models (DSMs). In these models feature vectors are created for each word, usually made up of context words with weights, and the similarity or relatedness of words is then calculated using vector similarity measures. Although DSMs have many possible parameters, a truly comprehensive study of these parameters, also fully considering the dependencies between them, is still missing and would be needed.

Most papers presenting DSMs focus on only one or two aspects of the problem, and take all other parameters as granted with some standard setting. For example, the majority of studies simply use cosine as vector similarity measure and/or (positive) pointwise mutual information as weighting scheme

ISSN 1135-5948. DOI 10.26342/2020-64-16

out of convention. Further, even in case of the considered parameters, usually only a handful of possible settings are tested for. Moreover, there are also such parameters that are completely ignored by most studies and have not been truly studied in the past, not even separately (e.g. smoothing, vector normalization or minimum feature frequency). What's more, as these parameters can influence each other greatly, evaluating them separately, one-by-one, would not even be sufficient, as that would not account for the interaction between them.

There are a couple of studies that consider several parameters for DSMs with multiple possible settings, but even these are far from truly comprehensive, and do not fully test for the interaction between the different parameters. So, although a comprehensive analysis of the possible parameters and their combinations would be crucial, there has been no such research to date. Further, although the best settings for the parameters can differ for different languages, the vast majority of papers consider DSMs for only one language (mostly English), or consider multiple languages but without a real comparison of findings across languages. Our study aims to address these gaps.

1.2 Aims and objectives

DSMs have two distinct phases in general. First, statistical distributional information (e.g. raw counts) is extracted from raw data (e.g. a large corpus). Then, feature vectors are created for words from the extracted information, and the similarity or relatedness of words is calculated by comparing their feature vectors. In our study we take the distributional information extracted in the first phase as already granted, and present a systematic study simultaneously testing all important aspects of the creation and comparison of feature vectors in DSMs, also caring for the interaction of the different parameters.

We have chosen to only study the second phase of DSMs, as the two phases are relatively independent from each other, and testing for every possible combination of parameter settings in the second phase is already unfeasible due to the vast number of combinations. So, instead of a full analysis, we already had to use a heuristic approach. Thus, we have omitted the examination of the first phase, as that would have been unreasonable and unmanageable, with one exception.

DSMs relying on information extracted from static corpora have two major categories, based on the type of their first phase: count-vector-based (CVBM) and predictive models (PM; also called word embeddings). In order to get a more complete view and due to the recent popularity of predictive models, in addition to using information extracted by a CVBM, we have also conducted experiments with information extracted by a PM for English. Further, we have also extended our analysis with a model based on a knowledge graph. Our intuition was that there will be a single configuration that achieves the best results in case of all types of models. However, please note that in the latter two cases only a part of the considered parameters could be tested due to the characteristics of such models. We have mainly focused on count-vector-based DSMs partly due to this.

During our research we have identified altogether 10 important parameters for the second phase of count-vector-based DSMs, such as vector similarity measures, weighting schemes, feature transformation functions, smoothing and dimensionality reduction techniques. However, only 4 of these parameters are available when predictive or knowledge-graph-based semantic vectors are used as input, as in those cases the raw counts are not available any more, the weighted vectors are already constructed and their dimensions are usually also reduced.

In the course of our analysis we have simultaneously evaluated each parameter with numerous settings in order to try to find the best possible configuration achieving the highest performance on standard test datasets. We have done our extensive analysis for English, Spanish and Hungarian separately, and then compared our findings for the different languages.

While also testing the conventionally used settings for each parameter, we also proposed numerous new variants in case of some parameters. Therefore, for many parameters a large number of settings (more than a thousand in some cases) were tested, resulting in trillions of possible combinations. All in all, we have considered a vast number of novel configurations, with some of these considerably outperforming the standard configurations that are conventionally used, and thus achieving state-of-the-art results.

First we have done our analysis for English and evaluated the results extensively (Dobó and Csirik, 2019a). Then we have repeated our analysis, with an increased number of settings for several parameters, for English, Spanish and Hungarian, and compared the findings across the different languages (Dobó and Csirik, 2019b).

2 Thesis overview

The thesis (Dobó, 2019) is organized into nine chapters, followed by the Appendices.

Chapter 1 gives an introduction to the topic, and presents our motivations and aims. Then we provide the theoretical background to the topic in Chapter 2, after which Chapter 3 introduces the used data and evaluation methods. Dobó and Csirik (2012) and Dobó and Csirik (2013) describe the inception of our research into the topic, and introduce a part of the used data and evaluation methods.

Chapter 4 is devoted to the detailed description of our analysis (Dobó and Csirik, 2019a), including the presentation of our two-phase heuristic approach and the description of the ten tested parameters. Our novel smoothing technique (Dobó, 2018) tested during our analysis is also presented here.

In Chapter 5 we present the results of our two-phase heuristic analysis for English, and then evaluate these results (Dobó and Csirik, 2019a). This is followed by the detailed comparison of the results for English, Spanish and Hungarian (Dobó and Csirik, 2019b) in Chapter 6.

We draw our conclusions in Chapter 7, which is followed by the English and Hungarian synopsis of the thesis in Chapters 8 and 9, respectively.

In the Appendices, we present the most important vector similarity measures and weighting schemes in detail, together with their formula. We also publish our novel general test datasets for Hungarian here, which, given the lack of previous such datasets, made the evaluation and comparison of our semantic models for Hungarian possible.

3 Main contributions

We have presented a very detailed and systematic analysis of the possible parameters used during the creation and comparison of feature vectors in distributional semantic models, for English, Spanish and Hungarian, filling a serious research gap. We have identified 10 important parameters of count-vector-based models and 4 relevant ones in case of using semantic vectors as input, and tested numerous settings for all of them. The main contributions of our work are as follows:

- Our analysis included novel parameters and novel parameter settings, and tested all parameters simultaneously, thus also taking their interaction into account. To our best knowledge, we are the first to do such a detailed analysis for these parameters, and also to do such an extensive comparison of them across multiple languages.
- Our novel two-step heuristic approach made the search for the best configurations among the numberless possibilities feasible for all three languages, and thus we were able to find such novel ones, many of them also incorporating novel parameter settings, that significantly outperformed conventional configurations.
- Although we had to use a heuristic approach due to the vast size of the search space, we have been able to verify the validity of this approach and the reliability and soundness of its results.
- While we have found that different configurations are best in case of models with count-vector-based, predictive and knowledge-graph-based semantic vectors as input, we have verified that a configuration performing well on given input data also works well on other input data of the same type.
- In accordance with our intuition, there were several parameters that worked very similarly in case of all three languages. We also found such parameters that were alike for Spanish and Hungarian, and different for English, which we also anticipated. However, it was interesting to see that there was such a parameter that worked similarly for English and Hungarian, but not for Spanish, and that we did not find any parameters that worked similarly for the two Indo-European languages, but differently for Hungarian.
- Although we have found that the very best results are produced by differ-

ent configurations for the different languages, our cross-language tests showed that all of them work rather well for all languages. Based on this we think that we could find such configurations that are rather language-independent, and give robust and reliable results.

- To be able to compare our results with the previous state-of-the-art, we have run such tests where the same data was used as input for both the previous state-of-the-art configurations and our configurations. In case of using raw counts as input and thus being able to optimize all 10 of our examined parameters, our best configurations contained novel parameter settings and clearly outperformed previous state-of-the-art configurations, with a considerable margin in most cases. When using semantic vectors as input and thus only being able to optimize 4 out of 10 parameters, our best configurations, also incorporating novel parameter settings, performed at least as well as the previous state-of-the-art, with a slight superiority in a couple of cases. All in all, our best model actually achieved absolute state-of-the-art results compared to all previous models of any type on the most important test datasets. Based on these results we think that our analysis was successful, and we were able to present such new parameter settings and new configurations that are superior to the previous state-of-the-art.

As it could be seen, the size of the input corpus, as well as the used information extraction method greatly influences the results. Therefore we think that doing an analysis similar to our current one for the information extraction phase of DSMs would be a principal direction for future research. Further, in our opinion it would be important to test our proposed new configurations using corpora magnitudes larger than that we could use. It would be even better if our whole heuristic analysis could also be repeated on these huge corpora. Further, although our results seem rather robust and reliable for Spanish and Hungarian too, it would be interesting to redo our analysis on larger and more reliable Spanish and Hungarian datasets, when such datasets will become available in the future.

We think that with this study we significantly contributed to the better understanding of the working and properties of DSMs. Although fully reliable conclusions from our results can only be drawn with respect to DSMs, we think that similar conclusions would hold for other NLP and non-NLP systems based on vector space models too.

For reproducibility and transparency, we have made our code and our most important resources publicly available at: https://github.com/doboandras/dsm-parameter-analysis/.

References

Dobó, A. 2018. Multi-D Kneser-Ney Smoothing Preserving the Original Marginal Distributions. *Research in Computing Science*, 147(6):11–25.

Dobó, A. 2019. *A comprehensive analysis of the parameters in the creation and comparison of feature vectors in distributional semantic models for multiple languages.* Ph.D. thesis, University of Szeged.

Dobó, A. and J. Csirik. 2012. Magyar és angol szavak szemantikai hasonlóságának automatikus kiszámítása. In *IX. Magyar Számítógépes Nyelvészeti Konferencia*, pages 213–224, Szeged, Hungary.

Dobó, A. and J. Csirik. 2013. Computing semantic similarity using large static corpora. In *39th International Conference on Current Trends in Theory and Practice of Computer Science*, pages 491–502, Špindlerův Mlýn, Czech Republic.

Dobó, A. and J. Csirik. 2019a. A comprehensive study of the parameters in the creation and comparison of feature vectors in distributional semantic models. *Journal of Quantitative Linguistics.*

Dobó, A. and J. Csirik. 2019b. Comparison of the best parameter settings in the creation and comparison of feature vectors in distributional semantic models across multiple languages. In *15th IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 487–499, Hersonissos, Greece.

Dobó, A. and S. G. Pulman. 2011. Interpreting noun compounds using paraphrases. *Procesamiento del Lenguaje Natural*, 46:59–66.

Procesamiento del Lenguaje Natural, Revista nº 64, marzo de 2020, pp. 131-134 recibido 25-11-2019 revisado 12-12-2019 aceptado 14-01-2020

Análisis de estructuras temporales en euskera y creación de un corpus

Analysis of Basque temporal constructions and the creation of a corpus

Begoña Altuna
begona.altuna@ehu.eus
Grupo Ixa, Universidad del País Vasco / Euskal Herriko Unibertsitatea (UPV/EHU)
Manuel Lardizabal 1, 20018 Donostia

Resumen: Tesis titulada "Euskarazko denbora-egituren azterketa eta corpusaren sorrera / Analysis of Basque temporal constructions and the creation of a corpus", defendida por Begoña Altuna Díaz en la Universidad del País Vasco (UPV/EHU) y elaborada bajo la dirección de las doctoras Arantza Díaz de Ilarraza (Departamento de Lenguajes y Sistemas Informáticos) y María Jesús Aranzabe (Departamento de Lengua Vasca y Comunicación). La defensa se celebró el 21 de noviembre de 2018 en la Facultad de Informática (UPV/EHU) en San Sebastián ante el tribunal formado por Kepa Sarasola (Presidente, Universidad del País Vasco (UPV/EHU)), Itziar Aduriz (Secretaria, Universidad de Barcelona (UB)) y Ricardo Etxepare (Vocal, Centre National de Recherche Scientifique (CNRS)). La tesis obtuvo la calificación de sobresaliente Cum Laude otorgada por unanimidad y mención internacional.
Palabras clave: Información temporal, euskera, extracción de información, corpus anotado, cronologías

Abstract: Ph. D. thesis entitled "Euskarazko denbora-egituren azterketa eta corpusaren sorrera / Analysis of Basque temporal constructions and the creation of a corpus", defended by Begoña Altuna Díaz at the University of the Basque Country (UPV/EHU) under the supervision of Dr. Arantza Díaz de Ilarraza (Languages and Computer Systems Department) and Dr. María Jesús Aranzabe (Basque Language and Communication Department). The thesis defense was held on the 21st of November 2018 at the Computer Science Faculty (UPV/EHU) in San Sebastian and the members of the commission were Dr. Kepa Sarasola (President, University of Basque Country (UPV/EHU)), Dr. Itziar Aduriz (Secretary, Universidad de Barcelona (UB)) and Dr. Ricardo Etxepare (Vocal, Centre National de Recherche Scientifique (CNRS)). The thesis was awarded an excellent grade and Cum Laude honours and the international mention.
Keywords: Temporal information, Basque, information extraction, annotated corpus, timelines

1 Introducción de la tesis

La información temporal ayuda a entender el contenido de los textos porque ayuda a ordenar las acciones y situaciones que se narran a lo largo del eje temporal. En el procesamiento del lenguaje natural, se han desarrollado esquemas de anotación (Pustejovsky et al., 2003a), corpus anotados (Pustejovsky et al., 2003b) y sistemas de identificación y normalización de información temporal (Strötgen y Gertz, 2013; Llorens, Saquete, y Navarro, 2010) para la interpretación automática de la misma para un sinfín de lenguas, pero no para el euskera, hasta el momento.

El objetivo de la tesis es generar los recursos y herramientas necesarios para el procesamiento de la información temporal en euskera. Para ello se han definido los siguientes objetivos parciales:

- Análisis de las estructuras que expresan información temporal en euskera y el tipo de información que representan.
- Desarrollo de un lenguaje de marcado

ISSN 1135-5948. DOI 10.26342/2020-64-17

para la información temporal en euskera.

- Creación de un corpus etiquetado de noticias y textos de historia para la experimentación.
- Creación de herramientas para la extracción y normalización de la información temporal.
- Desarrollo de una herramienta de generación de cronologías.

El trabajo de tesis se ha desarrollado en el grupo Ixa (UPV/EHU) y ha dado como fruto el análisis de la información temporal en euskera y el desarrollo de herramientas para el tratamiento automático de la misma.

2 Estructura de la tesis

La tesis se ha presentado en dos volúmenes, uno principal en euskera con título *Euskarazko denbora-egituren azterketa eta corpusaren sorrera* y una versión completa pero reducida en inglés con título *Analysis of Basque temporal constructions and the creation of a corpus*. Ambas comparten la misma estructura: la tesis se divide en cuatro partes principales, i) introducción, ii) etiquetado y creación del corpus, iii) herramientas y aplicaciones y iv) conclusiones y trabajos futuros, que se materializan en los siguientes capítulos.

1. En el capítulo introductorio se presenta el tema a investigar, la motivación para el mismo y los objetivos generales del trabajo.
2. En el segundo capítulo se presentan los trabajos realizados previamente en el procesamiento de la información temporal. Se describen brevemente los trabajos teóricos más relevantes, así como los recursos y sistemas que se han desarrollado para el procesamiento de la información temporal. Más concretamente, se presentan i) los lenguajes de marcado creados para diferentes tareas, ii) los corpus que contienen información temporal y las herramientas e interfaces para el etiquetado manual de los mismos, iii) las herramientas de extracción y normalización de la información temporal y iv) las herramientas avanzadas que toman como base información temporal estructurada.
3. En el tercer capítulo se analiza el modo en el que se expresa la información temporal en euskera. Se describen las principales estructuras temporales y las relaciones que se crean entre ellas.
    - **Eventos**: acciones, procesos, estados y predicaciones genéricas.
    - **Expresiones temporales**: estructuras textuales que expresan puntos e intervalos de tiempo.
    - **Relaciones aspectuales**: relaciones que expresan la fase del evento subordinado.
    - **Relaciones de subordinación**: relaciones entre dos eventos en las que uno es la cabeza y el otro es el subordinado.
    - **Relaciones temporales**: relaciones de orden cronológico entre dos eventos, dos expresiones temporales o entre ambos.

    Además, se identifican la información temporal que conlleva cada elemento y las características lingüísticas que expresan la misma.
4. En el cuarto capítulo se desgrana el proceso de creación del corpus EusTimeBank. EusTimeBank se ha etiquetado siguiendo EusTimeML, el lenguaje de marcado inspirado en TimeML (Pustejovsky et al., 2003a) para la información temporal en euskera, por medio del cual se ha codificado la información temporal identificada en el capítulo anterior.

    EusTimeBank está formado por 164 documentos (más de 73.000 tokens) que se dividen en tres subcorpus:
    - **FaCor**: 25 documentos originalmente en euskera sobre el cierre de Fagor.
    - **WikiWarsEU**: versiones en euskera de 19 narraciones históricas de WikiWars (Mazur y Dale, 2010).
    - **EusMEANTIME**: traducción a nivel de oración de los 120 documentos de MEANTIME (Minard et al., 2016) (noticias de economía).

    60 de esos documentos (51 de EusMEANTIME y 9 de FaCor) se han utilizado para el desarrollo y evaluación de las herramientas para el procesamiento de la información temporal en euskera (EusHeidelTime (Altuna, Aranzabe,

y Díaz de Ilarraza, 2017), bTime (Salaberri Izko, 2017) y KroniXa).

Las anotaciones se han hecho manualmente, para lo que se han desarrollado una directrices de anotación (Altuna, Aranzabe, y Díaz de Ilarraza, 2014b; Altuna, Aranzabe, y Díaz de Ilarraza, 2016). Tanto las anotaciones como las directrices han sido evaluadas en diferentes experimentos (Altuna, Aranzabe, y Díaz de Ilarraza, 2014a; Altuna, Aranzabe, y Díaz de Ilarraza, 2018a; Altuna, Aranzabe, y Díaz de Ilarraza, 2018b) en los que se ha medido el acuerdo entre anotadores en la identificación de las estructuras temporales y sus atributos y la idoneidad y corrección de las directrices.

5. En el quinto capítulo se describen las herramientas para el procesamiento de la información temporal en euskera que se han desarrollado a lo largo de la tesis.

    - EusHeidelTime es una herramienta basada en reglas para la identificación y normalización de expresiones temporales en euskera. Es la versión para euskera de HeidelTime (Strötgen y Gertz, 2013), del cual se ha adaptado el código fuente. Así, para el euskera, se han creado los recursos lingüísticos necesarios (reglas, patrones y valores normalizados) y se ha podido integrar la herramienta en la cadena de procesamiento del euskera (Otegi et al., 2016).
    - KroniXa toma la información extraída por EusHeidelTime y bTime, y usa las dependecias sintácticas para crear relaciones temporales dentro de las oraciones, para crear cronologías. KroniXa ordena los eventos anclándolos a los puntos de tiempo en los que suceden.

6. En el sexto capítulo se describen las contribuciones y conclusiones de la investigación y se presentan los trabajos futuros en el procesamiento de la información temporal en euskera.

3 Contribuciones de la tesis

En la tesis se ha abarcado el procesamiento de la información temporal en euskera de manera integral. Por un lado, se han creado los recursos lingüísticos para el procesamiento de la información temporal en euskera:

- Se ha analizado qué elementos transmiten información temporal (eventos y expresiones temporales), qué tipo de información transmiten y las relaciones que se crean entre ellos. Además, se ha analizado la información sobre la factualidad de los eventos.
- Se ha creado el lenguaje de etiquetado EusTimeML para anotar la información temporal en euskera. Para ello, se han definido las etiquetas, atributos y valores de los atributos. Se ha mantenido un esquema lo más parecido posible a TimeML para poder hacer comparaciones, pero se han hecho modificaciones en los valores de los atributos, para poder representar las características del euskera. También se han añadido atributos para poder representar la información de factualidad.

  Se ha evaluado la calidad las directrices de anotación mediante varios experimentos de etiquetado manual en los que se ha medido el nivel de acuerdo entre anotadores. Esto ha servido para aclarar y corregir las directrices que se han usado para anotar el corpus.
- Se ha creado el corpus EusTimeBank, que contiene 164 documentos de los que 60 se usan como *gold standard* para el entrenamiento y evaluación de las herramientas de extracción de información temporal. Se puede acceder libremente a los documentos en formato NAF[1].

Asimismo, se han desarrollado las herramientas para procesar la información temporal:

- Se ha desarrollado EusHeidelTime, la herramienta para la extracción y normalización de expresiones temporales. Se han creado las reglas, patrones y normalizaciones para el euskera y se ha conseguido una tasa de identificación (F1) de más del 80 % para la identificación total y del 90 % para la identificación parcial.
- La información temporal y las herramientas desarrolladas han servido como base para la creación de KroniXa y los recursos para su entrenamiento y evaluación, que están en pleno desarrollo.

[1] http://ixa2.si.ehu.es/eusheideltime

Agradecimientos

La tesis se ha desarrollado gracias a las becas PRE_2013_1_959, PRE_2014_2_242, PRE_2015_4_0284 y PRE_2016_2_294 del Gobierno Vasco y ha sido financiada por los proyectos PROSA-MED (TIN2016-77820-C3-1-R) del Ministerio Economía y Competitividad y DETEAMI (2014111003) del Gobierno Vasco.

Bibliografía

Altuna, B., M. J. Aranzabe, y A. Díaz de Ilarraza. 2014a. Euskarazko denbora-egiturak. Azterketa eta etiketatze-esperimentua. *Linguamática*, 6(2):13–24.

Altuna, B., M. J. Aranzabe, y A. Díaz de Ilarraza. 2014b. Euskarazko denbora-egiturak etiketatzeko gidalerroak (upv/ehu/lsi/tr;01-2014). Informe técnico, Ixa Group, University of the Basque Country.

Altuna, B., M. J. Aranzabe, y A. Díaz de Ilarraza. 2016. Euskarazko denbora-egiturak etiketatzeko gidalerroak v2.0 (upv/ehu/lsi/tr;01-2016). Informe técnico, Ixa Group, University of the Basque Country.

Altuna, B., M. J. Aranzabe, y A. Díaz de Ilarraza. 2017. EusHeidelTime: Time Expression Extraction and Normalisation for Basque. *Procesamiento del Lenguaje Natural*, 59(0):15–22.

Altuna, B., M. J. Aranzabe, y A. Díaz de Ilarraza. 2018a. Adapting TimeML to Basque: Event Annotation. En *Computational Linguistics and Intelligent Text Processing*, páginas 565–577, Cham, Switzerland. Springer International Publishing.

Altuna, B., M. J. Aranzabe, y A. Díaz de Ilarraza. 2018b. An Event Factuality Annotation Proposal for Basque. En *Proceedings of the Second Workshop on Corpus-Based Research in the Humanities,CRH-2*, volumen 1, páginas 15–24. Gerastree Proceedings.

Llorens, H., E. Saquete, y B. Navarro. 2010. TIPSem (English and Spanish): Evaluating CRFs and Semantic Roles in TempEval-2. En *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, páginas 284–291, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mazur, P. y R. Dale. 2010. WikiWars: A New Corpus for Research on Temporal Expressions. En *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, páginas 913–922, Stroudsburg, PA, USA. Association for Computational Linguistics.

Minard, A.-L., M. Speranza, R. Urizar, B. Altuna, M. van Erp, A. Schoen, y C. van Son. 2016. MEANTIME, the NewsReader Multilingual Event and Time Corpus. En *Proceedings of LREC 2016, Tenth International Conference on Language Resources and Evaluation*. European Language Resources Association.

Otegi, A., N. Ezeiza, I. Goenaga, y G. Labaka. 2016. A Modular Chain of NLP Tools for Basque. En *Proceedings of the 19th International Conference on Text, Speech and Dialogue, TSD 2016*, páginas 93–100, Cham, Switzerland. Springer International Publishing.

Pustejovsky, J., J. M. Castaño, R. Ingria, R. Saurí, R. J. Gaizauskas, A. Setzer, G. Katz, y D. R. Radev. 2003a. TimeML: Robust Specification of Event and Temporal Expressions in Text. *New directions in question answering*, 3:28–34.

Pustejovsky, J., P. Hanks, R. Saurí, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro, y M. Lazo. 2003b. The TimeBank Corpus. En D. Archer P. Rayson A. Wilson, y T. McEnery, editores, *Proceedings of Corpus Linguistics 2003*, numero 16, páginas 647–656, Lancaster, UK. UCREL, Lancaster University.

Salaberri Izko, H. 2017. *Rol semantikoen etiketatzeak testuetako espazio-denbora informazioaren prozesamenduan daukan eraginaz.* Ph.D. tesis, University of the Basque Country, Donostia.

Strötgen, J. y M. Gertz. 2013. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 47(2):269–298.

Información General

SEPLN 2020

XXXVI CONGRESO INTERNACIONAL DE LA SOCIEDAD ESPAÑOLA PARA EL PROCESAMIENTO DEL LENGUAJE NATURAL

Universidad de Málaga – Málaga (España)
23-25 de septiembre 2020
http://www.sepln.org/ y http://sepln2020.sepln.org/

1 *Presentación*

La XXXVI edición del Congreso Internacional de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN) se celebrará los días 23, 24 y 25 de septiembre de 2020 en la Escuela Técnica Superior de Ingeniería Informática (E.T.S.I. Informática) de la Universidad de Málaga.

La ingente cantidad de información disponible en formato digital y en las distintas lenguas que hablamos hace imprescindible disponer de sistemas que permitan acceder a esa enorme biblioteca que es Internet de manera cada vez más estructurada.

En este mismo escenario, hay un interés renovado por la solución de los problemas de accesibilidad a la información y de mejora de explotación de la misma en entornos multilingües. Muchas de las bases formales para abordar adecuadamente estas necesidades han sido y siguen siendo establecidas en el marco del procesamiento del lenguaje natural y de sus múltiples vertientes: Extracción y recuperación de información, Sistemas de búsqueda de respuestas, Traducción automática, Análisis automático del contenido textual, Resumen automático, Generación textual y Reconocimiento y síntesis de voz.

2 *Objetivos*

El objetivo principal del congreso es ofrecer un foro para presentar las últimas investigaciones y desarrollos en el ámbito de trabajo del Procesamiento del Lenguaje Natural (PLN) tanto a la comunidad científica como a las empresas del sector. También se pretende mostrar las posibilidades reales de aplicación y conocer nuevos proyectos I+D en este campo.

Además, como en anteriores ediciones, se desea identificar las futuras directrices de la investigación básica y de las aplicaciones previstas por los profesionales, con el fin de contrastarlas con las necesidades reales del mercado. Finalmente, el congreso pretende ser un marco propicio para introducir a otras personas interesadas en esta área de conocimiento

3 *Áreas Temáticas*

Se anima a grupos e investigadores a enviar comunicaciones, resúmenes de proyectos o demostraciones en alguna de las áreas temáticas siguientes, entre otras:

- Modelos lingüísticos, matemáticos y psicolingüísticos del lenguaje.
- Desarrollo de recursos y herramientas lingüísticas.
- Gramáticas y formalismos para el análisis morfológico y sintáctico.
- Semántica, pragmática y discurso.
- Resolución de la ambigüedad léxica.
- Generación textual monolingüe y multilingüe.
- Traducción automática.
- Síntesis del habla.
- Sistemas de diálogo.
- Indexado de audio.
- Identificación idioma.
- Extracción y recuperación de información monolingüe y multilingüe.
- Sistemas de búsqueda de respuestas.
- Evaluación de sistemas de PLN.

ISSN: 1135-5948

- Análisis automático del contenido textual.
- Análisis de sentimientos y opiniones.
- Análisis de plagio.
- Minería de texto en blogosfera y redes sociales.
- Generación de Resúmenes.
- PLN para la generación de recursos educativos.
- PLN para lenguas con recursos limitados.
- Aplicaciones industriales del PLN.

4 Formato del Congreso

La duración prevista del congreso será de tres días, con sesiones dedicadas a la presentación de artículos, pósteres, proyectos de investigación en marcha y demostraciones de aplicaciones. Además, prevemos la organización de talleres-workshops satélites para el día 25 de septiembre.

5 Comité ejecutivo SEPLN 2020

Presidente del Comité Organizador

- Eugenio Martínez Cámara (Universidad de Granada).

Colaboradores

- Flor Miriam Plaza del Arco (Universidad de Jaén).
- Pilar López Úbeda (Universidad de Jaén).
- Luis Alfonso Ureña López (Universidad de Jaén).
- Patricio Martínez Barco (Universidad de Alicante).
- Paloma Martínez Fernández (Universidad Carlos III)
- Francisco Javier Hortelano Ruiz (Universidad de Jaén)

6 Consejo Asesor

Miembros:

- Manuel de Buenaga Rodríguez (Universidad de Alcalá, España).
- Sylviane Cardey-Greenfield (Centre de recherche en linguistique et traitement automatique des langues, Lucien Tesnière. Besançon, Francia).
- Irene Castellón Masalles (Universidad de Barcelona, España).
- Arantza Díaz de Ilarraza (Universidad del País Vasco, España).
- Antonio Ferrández Rodríguez (Universidad de Alicante, España).
- Alexander Gelbukh (Instituto Politécnico Nacional, México).
- Koldo Gojenola Galletebeitia (Universidad del País Vasco, España).
- Xavier Gómez Guinovart (Universidad de Vigo, España).
- José Miguel Goñi Menoyo (Universidad Politécnica de Madrid, España).
- Bernardo Magnini (Fondazione Bruno Kessler, Italia).
- Nuno J. Mamede (Instituto de Engenharia de Sistemas e Computadores Investigação e Desenvolvimento em Lisboa, Portugal).
- M. Antònia Martí Antonín (Universidad de Barcelona, España).
- M. Teresa Martín Valdivia (Universidad de Jaén, España).
- Patricio Martínez Barco (Universidad de Alicante, España).
- Eugenio Martínez Cámara (Universidad de Granada, España).
- Paloma Martínez Fernández (Universidad Carlos III, España).
- Raquel Martínez Unanue (Universidad Nacional de Educación a Distancia, España).
- Ruslan Mitkov (University of Wolverhampton, Reino Unido).
- Leonel Ruiz Miyares (Centro de Lingüística Aplicada de Santiago de Cuba, Cuba).
- Manuel Montes y Gómez (Instituto Nacional de Astrofísica, Óptica y Electrónica, México).
- Lluis Padró Cirera (Universidad Politécnica de Cataluña, España).
- Manuel Palomar Sanz (Universidad de Alicante, España).
- Ferrán Pla (Universidad Politécnica de Valencia, España).
- Germán Rigau Claramunt (Universidad del País Vasco, España).
- Horacio Rodríguez Hontoria (Universidad Politécnica de Cataluña, España).
- Emilio Sanchís (Universidad Politécnica de Valencia, España).
- Kepa Sarasola Gabiola (Universidad del País Vasco, España).
- Thamar Solorio (University of Houston, Estados Unidos de América).
- Maite Taboada (Simon Fraser University, Canadá).
- Mariona Taulé (Universidad de Barcelona, España).

- Juan-Manuel Torres-Moreno (Laboratoire Informatique d'Avignon / Université d'Avignon, Francia).
- José Antonio Troyano Jiménez (Universidad de Sevilla, España).
- L. Alfonso Ureña López (Universidad de Jaén, España).
- Rafael Valencia García (Universidad de Murcia, España).
- René Venegas Velásques (Pontificia Universidad Católica de Valparaíso, Chile).
- M. Felisa Verdejo Maíllo (Universidad Nacional de Educación a Distancia, España).
- Manuel Vilares Ferro (Universidad de la Coruña, España).
- Luis Villaseñor-Pineda (Instituto Nacional de Astrofísica, Óptica y Electrónica, México).

7 Fechas importantes

Fechas para la presentación y aceptación de comunicaciones:

- Fecha límite para la entrega de comunicaciones: 2 de abril de 2020.
- Notificación de aceptación: 8 de mayo de 2020.
- Fecha límite para entrega de la versión definitiva: 22 de mayo de 2020.

Información para los Autores

Formato de los Trabajos

- La longitud máxima admitida para las contribuciones será de 8 páginas DIN A4 (210 x 297 mm.), incluidas referencias y figuras.
- Los artículos pueden estar escritos en inglés o español. El título, resumen y palabras clave deben escribirse en ambas lenguas.
- El formato será en Word o LaTeX

Envío de los Trabajos

- El envío de los trabajos se realizará electrónicamente a través de la página web de la Sociedad Española para el Procesamiento del Lenguaje Natural (http://www.sepln.org)
- Para los trabajos con formato LaTeX se mandará el archivo PDF junto a todos los fuentes necesarios para compilación LaTex
- Para los trabajos con formato Word se mandará el archivo PDF junto al DOC o RTF
- Para más información http://www.sepln.org/index.php/la-revista/informacion-para-autores

Información Adicional

Funciones del Consejo de Redacción

Las funciones del Consejo de Redacción o Editorial de la revista SEPLN son las siguientes:

- Controlar la selección y tomar las decisiones en la publicación de los contenidos que han de conformar cada número de la revista
- Política editorial
- Preparación de cada número
- Relación con los evaluadores y autores
- Relación con el comité científico

Funciones del Consejo Asesor

Las funciones del Consejo Asesor o Científico de la revista SEPLN son las siguientes:

- Marcar, orientar y redireccionar la política científica de la revista y las líneas de investigación a potenciar
- Representación
- Impulso a la difusión internacional
- Capacidad de atracción de autores
- Evaluación
- Composición
- Prestigio
- Alta especialización
- Internacionalidad

Cartas al director

Sociedad Española para el Procesamiento del Lenguaje Natural
Departamento de Informática. Universidad de Jaén
Campus Las Lagunillas, Edificio A3. Despacho 127. 23071 Jaén
secretaria.sepln@ujaen.es

Más información

Para más información sobre la Sociedad Española del Procesamiento del Lenguaje Natural puede consultar la página web http://www.sepln.org.
Si desea inscribirse como socio de la Sociedad Española del Procesamiento del Lenguaje Natural puede realizarlo a través del formulario web que se encuentra en esta dirección http://www.sepln.org/socios/inscripcion-para-socios/
Los números anteriores de la revista se encuentran disponibles en la revista electrónica: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/issue/archive
Las funciones del Consejo de Redacción están disponibles en Internet a través de http://www.sepln.org/category/revista/consejo_redaccion/
Las funciones del Consejo Asesor están disponibles Internet a través de la página http://www.sepln.org/home-2/revista/consejo-asesor/
La inscripción como nuevo socio de la SEPLN se puede realizar a través de la página http://www.sepln.org/socios/inscripcion-para-socios/