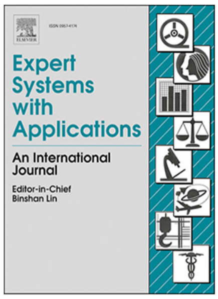
Journal Pre-proof



Exploiting discourse structure of traditional digital media to enhance automatic fake news detection

Alba Bonet-Jover, Alejandro Piad-Morffis, Estela Saquete, Patricio Martínez-Barco, Miguel Ángel García-Cumbreras

PII: S0957-4174(20)31027-7
DOI: https://doi.org/10.1016/j.eswa.2020.114340
Reference: ESWA 114340

To appear in: *Expert Systems With Applications*

Received date : 5 June 2020
Revised date : 22 October 2020
Accepted date : 16 November 2020

Please cite this article as: A. Bonet-Jover, A. Piad-Morffis, E. Saquete et al., Exploiting discourse structure of traditional digital media to enhance automatic fake news detection. *Expert Systems With Applications* (2020), doi: https://doi.org/10.1016/j.eswa.2020.114340.

Exploiting Discourse Structure of Traditional Digital Media to enhance Automatic Fake News Detection

Alba Bonet-Jover[a], Alejandro Piad-Morffis[b], Estela Saquete[a], Patricio Martínez-Barco[a], Miguel Ángel García-Cumbreras[c]

[a] *Department of Software and Computing Systems, University of Alicante, Spain*
{alba.bonet,stela,patricio}@dlsi.ua.es
[b] *School of Math and Computer Science, University of Havana, Cuba*
apiad@matcom.uh.cu
[c] *CEATIC / Universidad de Jaén*
magc@ujaen.es

Preprint submitted to ESWA *October 22, 2020*

Abstract

This paper presents a novel architecture for dealing with Automatic Fake News detection. The architecture factors in the discourse structure of news in traditional digital media and is based on two premises. First, fake news tends to mix true and false information with the purpose of confusing readers. Second, this research is focused on fake news delivered in traditional digital media, so our approach considers the influence of the journalistic structure of news, and the way journalists tend to introduce the essential content in a news story –using 5W1H answers–. Considering both premises, this proposal deals with the news components separately because some may be true or false, instead of considering the veracity value of the news article as a unit. A two-layer architecture is proposed, Structure and Veracity layers. To demonstrate the validity of the proposal, a new dataset was created and annotated with a new fine-grained annotation scheme (FNDeepML) that considers the different elements of the news document and their veracity. Due to the severity of the COVID-19 pandemic crisis, health is the chosen domain, and Spanish is the language used to validate the architecture, given the lack of research in this language. However, the proposal can be applied to any other language or domain. The performance of the Veracity layer of our proposal, which factors in the traditional news article structure and the 5W1H annotation, is capable of delivering a result of F_1=0.807. This represents a strong improvement when compared to the baseline, which uses the whole document with a single veracity value, obtaining F_1=0.605. These findings validate the suitability and effectiveness of our approach.

Keywords: Natural Language Processing, Fake News, Automated Fact-checking, Deep Learning, Machine Learning, Human Language Technologies

1. Introduction

In the digital era, information is mostly received and accessed online and the quality of this information becomes a crucial issue. However, there is a huge world-wide problem regarding the dissemination of fake news whose aim is to create confusion and manipulate public opinions and behaviours. Fake news are structured and written in a way that makes it difficult to distinguish between what is true or false. Fake information is diffused significantly farther, faster, deeper, and more broadly than the truth in all categories of information (Vosoughi et al., 2018).

This situation is exacerbated in times of emergency such as during the 2020 global pandemic caused by COVID-19. There are several reasons that have made coronavirus hoaxes a potentially serious problem. Information on COVID-19 was scarce during the early stages of the crisis, which increased the problem of misinformation. Besides, people around the world were in lockdowns, hyperconnected and anxious, which led to exponential viralization compared to a normal situation. Finally, many hoaxes related to prevention or cures were released, albeit with the intention of protecting, but these remedies spreading unchecked can be highly damaging. One example of false information widely disseminated was the claim *"Russia released more than 500 lions to make sure that people stay inside during the COVID-19 pandemic"*. The aim was to create alarm but it was demonstrated to be false[1].

In many cases, this disinformation is delivered by digital media web pages, which present news articles following the traditional format of a news piece, but sometimes "fake" information is provided, confusing readers and, in the case of fake news related to health, putting at serious risk the well-being of these people who may follow the advice given. Detecting and tackling fake news quickly and efficiently is, therefore, crucial because once false information spreads and permeates throughout society, it becomes difficult to refute. The number of hoaxes is reaching levels that would benefit from applying automatic techniques that enable the detection of fake news before they are massively spread. This is why Artificial Intelligence and Natural Language Processing (NLP) techniques are applied, so that the process can be automated.

A common phenomenon in the context of fake news is that false infor-

[1]https://www.snopes.com/fact-check/russia-release-lions-coronavirus/

mation is provided mixed with true information, to create confusion in the reader, and this premise is the basis of our proposal. An example is the claim *"U.S. President Donald Trump will benefit financially if hydroxychloroquine becomes an established treatment for COVID-19"*, which was fact-checked as mostly false. Furthermore, by studying the journalistic structure of news and how journalists introduce the essential content in news stories, our proposal considers the information as separated items, where some are true and some are false, instead of considering the news article as a whole when giving it a veracity value. This research proposal aims to help automatic learning systems to determine which parts of the structure of a news piece, or which type of content is more influential in reaching a decision about the veracity of the news (Conroy et al., 2015) (Pérez-Rosas et al., 2018). From hereafter, the term veracity refers to the accuracy and the truthfulness of the information provided in a traditional digital news document (Ciampaglia et al., 2015) (Das Bhattacharjee et al., 2017) (Lewandowsky et al., 2012) (Nyhan et al., 2012).

Considering the present context, the main contributions of this research are the following:

- Firstly, the proposal of a novel architecture for automatic fake news detection on traditional digital newspaper articles that can determine not only the full document veracity but most importantly, the veracity of the essential content elements of the news. The architecture will demonstrate that it is possible to determine the veracity of the news more accurately by taking advantage of the discourse structure of the news, that is, the journalistic structure and the essential content of the news piece, thereby reducing the noise when training automatic learning systems.

- Secondly, due to the lack of resources where information is annotated as independent parts, another important objective of this research is the creation of a dataset using a fine-grained annotation scheme, named FNDeepML. This annotation scheme is especially focused on differentiating the structural elements and essential content of classic news articles, which should respond to the 5W1H (What, When, Who, Where, Why and How) questions. This approach is especially innovative because existing datasets tag the news as a whole, in a single veracity category. The language chosen for the dataset is Spanish, because despite being the third most spoken language in the world[2], there are very few Spanish language resources for this task at the present time, making it beneficial for the research community. Due to the alarming pandemic situation, the health domain is used as a benchmark, but the proposal is readily adaptable to any language and domain.

The rest of the paper is organized as follows: Section 2 describes the structure of newspaper articles and their main content as well as the background of automatic fake news detection regarding NLP; Section 3 presents the definition of a new annotation scheme and the dataset created following this scheme; Section 4 shows the architecture of the automatic system proposed; Section 5 describes the evaluation environment used in this research; Section 6 shows the evaluation results and discusses them; Section 7 presents a set of experiments to compare our proposal with the state of the art (SOTA); and finally, our conclusions and future work are presented in Section 8.

2. Background

The development of automatic systems for fake news detection in the context of this proposal requires the analysis of the main features of newspaper articles, such as how they are structured and how the content is presented. It is important to focus on everything that can serve as a differentiating element between true news and fake news. Furthermore, a revision of the most relevant literature regarding computational mechanisms for automatic fake news detection is presented.

2.1. News structure and the 5W1H method

News is usually presented within a specific structure to attract readers and provide information in an interesting and organised way. Although there are different ways of writing a news story, there are two key principles on which all well-built news should be based: neutrality and the inverted pyramid structure (Thomson et al., 2008). Thus, the objectivity of a news piece may depend on these two factors, so the detection of unusual deviations from these accepted journalistic norms could provide a clue to detect fake news.

In the inverted pyramid hypothesis, "certain parts of news articles carry different levels of useful information" (Khan et al., 2018; Norambuena et al.,

[2]https://www.cervantes.es/imagenes/File/espanol_lengua_viva_2019.pdf

2020), placing the most important information first and ending with the least relevant information (Zhang & Liu, 2016). The three common and most important parts of the news structure are the headline, the lead and the body. Other important but secondary elements of news are the subtitle or the conclusion that usually appear in news articles, but they are not always present (see Figure 1).

In a well-built article, the parts must appear the following order:

- *Headline*: This element is the title of the news article and it provides the main idea of the story. Normally it summarizes, in one sentence, the basic and essential information about the story. Its main objective is to attract the reader's attention.

- *Subtitle*: A second title that explains the headline in a little more detail. It completes the information, but it also presents the idea in a very summarized way. Sometimes, it completes the information given in the headline, and at times it provides other details not mentioned before. Its function is to hold the reader's attention and to encourage him/her to keep reading the news article.

- *Lead*: The paragraph(s) that develops the main information by following the 5W1H method and "presents the point or newsworthy element(s) of the story and simultaneously works as a beginning of the story" (Bednarek & Caple, 2012). All the main information of the news article must be clearly presented in this section by answering the six questions used in journalism: what, who, where, when, why and how. The lead and the headline are sometimes considered as a unit because the lead usually repeats the idea given by the headline, but in more detail and accuracy (Thomson et al., 2008).

- *Body*: All the developed information is in this part of the news article. The body presents all the background, facts, elements and reasons of the story in detail. As mentioned by (Thomson et al., 2008), "the body of the text does not develop new meanings but, rather, acts to refer back to the headline/lead through a series of specifications." All the six questions answered in the lead will be developed in the body by explaining all the elements involved.

- *Conclusion/Tail*: The main idea of the story can be summarized in a phrase or in a paragraph, but, even if the conclusion is a part of a well-built article, it does not always appear. It does not present novel information, as it is only a summary.

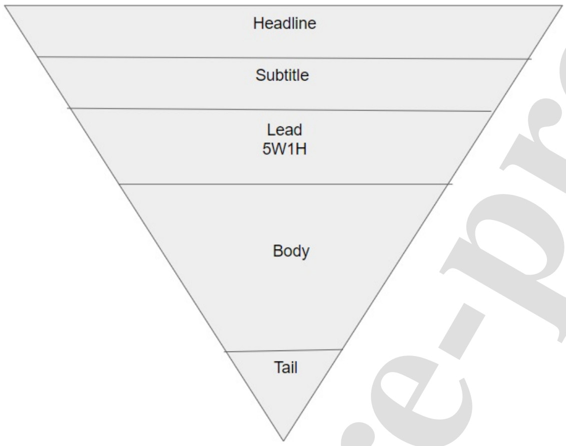


Figure 1: Inverted pyramid in newspaper articles

Besides the news structure, journalism purists argue that a story is not complete until the essential content is presented by answering six questions: `WHAT`, `WHO`, `WHERE`, `WHEN`, `WHY` and `HOW`. This method is known as 'five Ws and one H' (5W1H) (Chakma & Das, 2018a; Kim et al., 2012; Wang et al., 2010). Specifically, the six questions refer to:

- `WHAT`: The circumstances, the event, the facts.
- `WHO`: People involved in the events.
- `WHERE`: The location where the events occurred.
- `WHEN`: The time or the moment when the events occurred.
- `WHY`: The reason or the cause of the event.
- `HOW`: The way events have developed.

The 5W1H method is essential in the lead construction (Chagas, 2019). Besides, the lead is an essential part in a piece of news as it presents the main elements of an article: fact, actors, place, time, reason and manner, thus answering the six questions that are key to communicating a story accurately.

However, these questions are not always answered in the lead. It is sufficient that only the two or three most important questions are answered and the remaining questions will be answered in detail in the body of the news.

From a computational perspective, automatic extraction of 5W1H was applied to different tasks and languages, such as English or Chinese (NIST, 2011) (Hamborg et al., 2018) (Chakma & Das, 2018b) (Han et al., 2013) (Wang, 2012). These works demonstrated that the task is feasible, where for instance, GiveMe5W1H (Hamborg et al., 2018) are obtaining a mean average generalized precision 0.73 for all categories and 0.82 for 'who', 'what', 'when', and 'where' for English language. Despite these results being encouraging, as far as we know, those tools are not available in Spanish, and no other similar resource was found at this moment.

2.2. Fake News using NLP

Considering that digital information is disseminated exponentially, natural language processing and Machine Learning (ML) approaches play a fundamental role in fake news detection (Dale, 2017). Given that assessing the veracity of a news story is complex from an engineering point of view, the research community is approaching this task from different perspectives (Saquete et al., 2020).

Current fake news detection research has been conducted treating each news piece as a whole to be classified with a veracity category based on: lexical, syntactic and semantic content of the news as a whole (also known as content-based features); or, issues related to the user or viralization of the news (also known as context-based features)(Conroy et al., 2015).

Fake news detection currently focuses on studying linguistic aspects of falsehood by identifying different types of features for fake news. (Zhou & Zhang, 2008) proposed a system with the features classes, such as quantity (amount of information), language complexity, expressiveness, message content: n-grams, affect (positive or negative emotions), etc. (Pérez-Rosas et al., 2018) described a similar set of features, grouped by general categories, such as ngrams, punctuation, psycholinguistic features, readability and syntax. It is very common to use the Linguistic Inquiry and Word Count(LIWC) (Newman et al., 2003), which is a text analysis program that counts words in psychologically meaningful categories and is available in different languages. Using the Spanish language, (Almela et al., 2012) created an opinion dataset consisting of 200 assessments about different topics and tested the categories of LIWC.

Shloka Gilda (Gilda, 2017) demonstrated the relevance NLP to detect fake information. They used time period frequency-inverse record frequency (TFIDF) of bi-grams and probabilistic context free grammar (PCFG) detection. Very recent works like (Faustini & Covoes, 2020) proposed extracting text features to deal with the problem at a multilingual level.

Stylometry is the application of the study of linguistic style generally to written language. Regarding automatic Fake News detection, Potthast et al. (Potthast et al., 2018) used stylometry, combining writing style features such as n-grams, stop words, and parts-of-speech; and ones specific to the news domain, such as 10 readability scores and dictionary features, each indicating the frequency of words from a tailor-made dictionary in a document, using the General Inquirer Dictionaries as a basis. The domain-specific features include ratios of quoted words and external links, the number of paragraphs, and their average length. Afroz et al. (Afroz et al., 2012) also used stylometry to detect deception in online writing. More than 700 features were selected (lexical, syntactic, content specific, grammar and vocabulary complexity, uncertainty, etc). They used three feature sets to identify stylistic deception: i) Writeprints feature set (lexical, syntactic, and content specific); ii) Lying-detection feature set (such as q quantity, vocabulary complexity or specificity and expressiveness); iii) 9-feature set (authorship-attribution features), nine features that were used in the neural network experiments in Brennan's work (Brennan & Greenstadt, 2009). The main conclusion was that two kinds of adversarial attacks —imitation and obfuscation— can be detected with high accuracy using a large feature set. Non-content specific features have the same accuracy as content-specific features, and even by ignoring the contextual similarity of documents, it is possible to detect adversarial documents with sufficient accuracy. Furthermore, previous linguistic research has shown that the frequencies of common function words are content neutral and indicative of personal writing style (Mosteller & Wallace, 1963).

Regarding context features, Kai et al. (Shu et al., 2019) proposed a technique that exploits relationships among publishers, news pieces and users to predict fake news. They employ a linear classifier and assign each user a credibility score based on the user's online behavior. A low credibility score correlates to fake news.

Volkova et al. (Volkova et al., 2017) presented a technique that classifies suspicious posts by combining content and context features via the use of linguistic and network features.

Both Machine Learning (ML) and Deep Learning (DL) algorithms applied the previously mentioned content and context features and delivered similar results when tackling the problem of classifying the text. A summary of the most commonly used detection strategies are indicated below.

- Classification approaches based on **Machine Learning**: (Gravanis et al., 2019; Conroy et al., 2015; Rubin et al., 2016; Pérez-Rosas et al., 2018; Almela et al., 2012; Afroz et al., 2012; Shu et al., 2019; Chen & Chen, 2014; Mihalcea & Strapparava, 2009)

- Classification approaches based on **Deep Learning**: (Das Bhattacharjee et al., 2017; Volkova et al., 2017; Ren & Ji, 2017; Zhou & Zhang, 2008; Monti et al., 2019; Verma et al., 2019; Rashkin et al., 2017)

- Classification approaches based on **Ensemble Learning approaches**: Very recent works are not using a single ML or DL model to tackle the problem, but an ensemble learning approach (Agarwal & Dixit, 2020). Additionally, some approaches optimize the weights of the ensemble with an external technique, such as Self-Adaptive Harmony search (Huang & Chen, 2020).

- Other approaches: (Brennan & Greenstadt, 2009)

Most previously cited systems use ML as a detection system, and specifically SVM in most cases. It is true that in recent years systems based on LSTM and DL in general have been incorporated, which use open systems such as BERT (Devlin et al., 2018). From the analysis of the literature, combining linguistic features with ML or DL approaches obtains some interesting results, but they seem to reach the ceiling in terms of performance. This suggests that hybrid methodologies that combine these content approaches with context information could provide a strategy to enhance performance. In addition, often, the ML or DL approximations behave like black boxes which makes it difficult to explain the generated models. The use of ensemble learning can boost performance, especially when aggregating several low-performing models, or models with different hypothesis spaces. For example, one model based on linguistic features and another model based on external knowledge. Neural models are not often explicitly ensembled, since they already offer techniques to achieve the same effect (e.g., dropout).

2.2.1. Fake News datasets

The purpose of this section is to present the structure currently followed by the most relevant datasets that Fake News Detection systems are using. Given that our goal is to study what type of annotation is currently being used, we have analysed the datasets presented in the literature even though their language is mainly English.

To the authors' knowledge, current approaches are using datasets where the news article is classified as a whole with a veracity value. (Vlachos & Riedel, 2014) are the first to release a public fake news detection and fact-checking dataset that includes 221 statements. The statements were classified by using a five-point scale: true, mostlytrue, halftrue, mostlyfalse and false. After this, (Ferreira & Vlachos, 2016) have released the Emergent dataset. In this dataset, a set of claims are classified according to their veracity and the stance of articles mentioning these claims. (Pérez-Rosas et al., 2018) introduced two new fake news datasets, one obtained through crowdsourcing and covering six news domains, and another obtained from the web covering celebrities, and classified as Fake or Legitimate. BuzzFeedNews[3], is a dataset comprised of a sample of news published in Facebook from 9 news agencies over a week close to the 2016 U.S. election. The LIAR dataset was presented at (Wang, 2017). They collected 12.8K manually labeled short statements from various contexts spanning a decade. This dataset is larger than the previous largest public fake news datasets of a similar type. The news articles are usually classified using a veracity scale, from true to false (pants-fire, false, barelytrue, half-true, mostly-true, and true). But, again, the whole text is annotated with a category as an atomic unit. Kaggle Fake News dataset is provided by the Kaggle competition [4], which is is a popular platform with excellent resources for those who want to learn ML and even data science. The Kaggle dataset contains English fake and true news articles from 2015–2018. The dataset contains text and metadata from 244 websites and represents 12,999 posts in total.

Regarding Spanish datasets, Posadas et al. (Posadas-Durán et al., 2019) presented a Spanish dataset that contains 491 true news and 480 fake news items. Almela et al. (Almela et al., 2012) presented a Spanish dataset of

[3]available at https://github.com/BuzzFeedNews/2016-10-facebook-fact-check/tree/master/data

[4]available at https://www.kaggle.com/

three different topics: opinions on homosexual adoption, opinions on bull-fighting, and feelings about one's best friend. They collected 100 true and 100 false statements for each topic, with an average of 80 words per statement. Arguably, there is a shortage of resources in languages other than English (Silva et al., 2020), and specifically in Spanish. Besides, although some of the datasets are in Spanish, and some are even annotated with a classification based on graded nuances for truthfulness, as was the case with the English datasets, in all cases, the annotation is of the whole textual unit rather than the parts comprising it. Given previous research on the task, the novelty of the work presented here relies on an architecture that exploits a new fine-grained annotation[5] in a two-layer architecture. This allows the reduction of noise when training ML and DL systems. Even though the proposal is focused on Spanish and the health domain, it can be readily applied to different languages and domains.

3. A New Benchmark dataset for Spanish Fake News Detection

Next, the definition of the fine-grained annotation scheme, known as FN-DeepML, is presented, as well as the information about the dataset created using the said annotation scheme.

3.1. FNDeepML Annotation scheme

The annotation scheme applied to the dataset is able to distinguish the structure of the news piece, the essential parts within it and the characteristic elements that shape news. The scheme comprises two levels of representation:

1. *Newspaper article structure:* At this first level, the five elements of the newspaper article structure are annotated using one of these tags: `HEADLINE`, `SUBTITLE`, `LEAD`, `BODY` and `CONCLUSION`. In the case of the title and subtitle, it will almost always be the first two sentences, the lead is usually the first introductory paragraph of the news, the body usually corresponds to the remaining paragraphs of the news and the conclusion, as a rule, is the last paragraph or any concluding sentence. Furthermore, another tag has been defined at this first level: `QUOTE`. This tag could appear embedded in the previous elements. It

[5]The annotation is performed manually in training and automatically in testing

is used when an element or sentence textually quotes a message or reproduces an already reported idea.
For each tag, there is a numerical `id` attribute to identify each element; and a `type` attribute, that will indicate the value of truth or deception. These values will be indicated as follows: "T" (true text), "F" (fake text) or "U" (a text whose veracity is unknown). In this way, fake and true elements can be detected in the same news piece. In the case of the `QUOTE`, there is no `type` attribute but an attribute called `author_stance` whose possible values are: "D" (the author disagrees with the quote); "A" (the author agrees with the quote); and "U" (Unknown, if the author's stance is not clear). `QUOTE` is an element that differs from the basic inverted pyramid structure elements because it is only used to frame a set of external information –5W1H tags– with a veracity value that should not be learned by the system in the same way as the rest of 5W1Hs. This is due to the fact that it is information reported with which the author may or may not agree (depending on the `author_stance` value). So, the veracity value of the 5W1Hs within a `QUOTE` will be tuned by the author's stance during the training process. For that reason, the "type" attribute linked to the `QUOTE` tag is not required.

2. *Essential news content (5W1H):*
   In the second level of annotation, the essential content of the news piece is marked by annotating the answers to "the 5 Ws and the 1 H", using the following tags for each case: `WHO, WHEN, WHERE, WHAT, WHY` and `HOW`. All present 5W1H elements incorporated in the news piece were annotated. These tags have two mandatory attributes and one optional. All the 5W1H tags are annotated with the attributes `type`, with the same description as the first level tags; and `id` to determine if more than one content tag appears in the same news piece. For example, if there are two `WHO` items, if they refer to different people they would have a different `id` value. There is also an optional attribute, termed `not_relevant`, and this term is assigned a true value when the information provided by the 5W1H tag's content is not semantically relevant to determine the veracity of the news article. In order to annotate the 5W1H items, first, the different facts found in the text are detected, though understanding a fact from a given sentence means being able to answer "Who did what to whom, when, where, why, and how?". To answer such questions of who, what, etc., it is important

to identify each syntactic constituent of a sentence such as predicates, subjects, objects etc. Rules already defined in the literature to identify the answer to the 5W1H questions have been followed (Voorhees, 2001) (Hamborg et al., 2018) (Chakma & Das, 2018b) (Han et al., 2013) (NIST, 2011) . Semantic role labelling tools[6] were used to support manual annotation.

Moreover, metadata that are part of news content and provide information about the creation of news are the domain (`DOMAIN`), the source (`SOURCE`), the date (`DATE`), and the author (`AUTHOR`).

3.2. Dataset description

To create a Fake News dataset in Spanish (Bonet-Jover et al., 2020b), news documents in Spanish belonging to the health domain (topics such as COVID-19 which is a 50% of the dataset) were automatically collected[7]. To build the dataset in a balanced manner, fake and true news were collected from several online newspapers, blogs and fact-checking websites. The following news websites were used for collecting fake news, among others: Biosalud; Tengafe; Okdiario; Bioguia; Eje21; La Cháchara; Tudiario.net; Vidanaturalia; TICbeat; and, Acta sanitaria. For true news, websites such as the following were used among others: Kernpharma; Cuidateplus; Cinfasalud; Boticaria García; Comer o no comer; Julio Basulto; Nutrimedia; Vital; and, the press sections of official organizations' sites —The World Health Organization (WHO), La Asociación Española Contra el Cancer (AECC) , or the National Cancer Institute (NCI)—.

A total of 200 news documents were collected. More specific figures relating to the dataset built are presented in Table 1.

Type of News	**No Docs**	**No tokens**	**Avg tokens per doc**	**Avg tokens Headline**	**Avg tokens Lead**	**Avg tokens Body**
True News	105	75951	723	12	77	562
Fake News	95	58581	617	12	63	494
Total	200	134532	670	12	70	530

Table 1: General dataset description

[6]http://nlp.lsi.upc.edu/freeling/node/1
[7]Corpus download: https://doi.org/10.5281/zenodo.4090914

A manual annotation was carried out on the news collected, following the FNDeepML annotation scheme described in Section 3.1.

To ensure the veracity of news as well as that of the different 5W1H items, a manual cross-referencing information checking procedure was conducted using information from official websites like WHO and the fact-checks collected by Spanish fact-checking organizations belonging to the IFCN[8], such as Newtral[9], Salud sin Bulos[10], Maldita[11], Chequeado[12] or, AFP Factual[13]. Fact-checking agencies verify the information delivered in the different media in order to determine its veracity and correctness. They publish these fact-checks to make them available to the public. Furthermore, the online application entitled “Google Fact Check Explorer”[14] was also used to check the veracity of the information.

This procedure verifies the veracity category of each 5W1H by searching in these resources and determining if there is a previous fact-check where the 5W1H element is involved, whereby the corresponding category assigned to the fact-check would be assumed. If information does not appear in any of the sites mentioned above, we cannot determine the truthfulness or falseness and hence the category of Unknown is adopted. Determining the veracity category of each 5W1H element is dependent on their context and they would be classified as true or false depending on their relationship with other 5W1H elements, as well as the context in which the statement is included. For example, it is possible to have different veracity values of the same WHO. Take examples (1.a) and (1.b) where WHO=“Donald Trump” appears in different newspaper documents, and after the manual cross-referencing procedure, one was found True and the other was a hoax and assigned a False value.

(1) a. <WHO id=1 type='T'> Donald Trump </WHO> is the new candidate for US elections in 2020

b. <WHO id=1 type='F'> Donald Trump </WHO> discovers the COVID-19 vaccine

Table 2 presents the percentage and total items per document part clas-

[8]International Fact Checking Network (https://www.poynter.org/ifcn/) is a unit of the Poynter Institute dedicated to bringing together fact-checkers worldwide.

[9]https://www.newtral.es/

[10]https://saludsinbulos.com/

[11]https://maldita.es/

[12]https://chequeado.com/

[13]https://factual.afp.com/

[14]https://toolbox.google.com/factcheck/explorer

sified as True, False and Unknown of the whole dataset, following the previously defined annotation scheme that was carried out manually. The results indicate a very balanced dataset. Regarding the `QUOTE` tag, since this element does not have a veracity value, it is not included in the table, but there are 8 `QUOTE` in the false news part of the dataset, quoting statements that support the fake news, and 140 in the true news part of the dataset, which confirms that there is a high amount of refutations present in current true news.

Type	HEADLINE	SUBTITLE	LEAD	BODY	CONCLUSIONS
True	50.75%	52.22%	46.45%	53%	50.40%
False	45.27%	28.89%	33.88%	47%	33.60%
Unknown	3.98%	18.89%	19.67%	0%	16.00%
Total items	200	90	183	200	125

Table 2: Percentage and total number of items per document part classified as True, False and Unknown of the whole dataset

Each news piece was divided into the parts presented in Section 2.1 and the 5W1H found in the three top parts of the content (headline, subtitle and lead) were also marked. The experts were asked to mark the divided items with true, false or unknown[15] based on the fact-checks of the news. Details of the figures regarding 5W1H are shown in Table 3.

Type	WHAT	WHO	WHEN	WHERE	WHY	HOW
True	41.64%	0.13%	30.41%	39.67%	32.26%	42.72%
False	35.43%	0.26%	25.77%	19.33%	45.16%	38.35%
Unknown	22.93%	99.61%	43.81%	41.00%	22.58%	18.93%
Total items	1112	766	194	300	62	206

Table 3: Percentage and total number of items per type of question (5W1H) classified as True, False and Unknown of the whole dataset

Considering the false news part of the dataset, Table 4 presents the percentages of the different veracity values obtained for each of the news structure elements as well as for the different 5W1H items. This table only includes figures extracted from false news articles of the dataset excluding true news wherein all elements are true.

[15]The information provided was not fact-checked as true or false

Item	False (%)	True (%)	Unknown (%)	Total items
HEADLINE	**95.79**	0	4.21	95
SUBTITLE	68.42	10.53	21.05	38
LEAD	75.31	6.17	18.52	81
BODY	**100**	0	0	95
CONCLUSION	80.38	5.88	13.73	51
WHAT	**68.70**	6.11	25.18	409
WHERE	24.79	22.31	**52.89**	121
WHEN	43.02	9.30	**47.67**	86
WHO	0.59	0	**99.41**	340
WHY	**61.54**	15.38	23.08	39
HOW	**60.82**	16.49	22.68	97

Table 4: Distribution of false, true and unknown items found in the false news part of the dataset, excluding true news

After a manually analyzing the dataset and the figures presented in Table 4, some preliminary conclusions regarding the false part of the dataset were extracted:

- *Newspaper article structure*: The *headline* is practically always false in news documents detected as false. Obviously, the *body*, upon being annotated as a whole will be classified as false for all fake news. Furthermore, the *headline* and the *body* are presented in all news, but the *lead* is not always part of the false news structure.

- *The 5W1H*: *What* is the part where most false information is provided, although there is also a high degree of undefined information. The false information provided in the *Why* and *How* is also very high and close to the *What* values. In the case of *Who, When* and *Where* items, there is a high degree of vagueness, especially in *Who* items. Objective news provides accurate and concrete data, so detecting these inaccuracies enables us to determine if a news story is reliable. A few examples of vague *Who* tags are: *"los expertos"* ("the experts") or *"investigadores"* ("researchers"). These *Who* terms are generic, and not specific authors because fake news usually avoid revealing a specific source that would make the information reliable. Concerning *Where* tags, some of the imprecise examples are: *"en algunas ciudades"* ("in some towns") or *"en otros países"* ("in other countries"). In these cases, the examples indicated do not refer to a specific place and that makes the information imprecise. With regard to *When* tags, some vague expressions include:

"hace unos meses" ("some months ago") or *"en los próximos años"* ("in the next years"). Just like places, times are also ambiguous, so information is not reliable.

3.3. Dataset annotation task

The annotation of the dataset was first applied by an annotator with linguistic training in translation and interpretation. This annotator was the person in charge of compiling the dataset and implementing the annotation schema in news. For the annotation, an annotator with journalism training was also involved. In the first phase, a simple annotation of very few documents was done in order to train the two annotators on the guidelines (Bonet-Jover et al., 2020a). Once the first annotation was done, the quality of the annotation scheme was analyzed according to the annotation agreement, including only the items where both annotators coincided. Subsequently, a meeting was held to analyse the items with different annotations with the aim of arriving at a consensus. Afterwards, the modifications required were actioned, both in terms of the guidelines and the annotations.

In order to measure the quality of the dataset annotation, an inter-annotator agreement between two annotators using Cohen's *kappa* (Cohen, 1960) was performed obtaining k=0.737 for the 1st level of annotation categories and k=0.851 for the 2nd level of annotation categories, which validates the labeling.

As for the annotation time for the 200 news article dataset, 200 hours were employed (1 hour per document), 20 hours for correcting mistakes of the dataset and 30 hours for annotator training and comparison of annotations. In the case of disagreement, the annotators compared their annotations and reached a consensus, but these cases took approximately 30 extra hours to resolve, increasing the total time to complete the process to 280 hours.

4. Pandemic Fake News Detection system: Design and Development

A two-layer architecture based on a pipeline is proposed. The rationale is based on the hypothesis that the structural parts and essential content of a news piece have specific veracity values, which influence the overall veracity value of the news story. This can also be inferred from the conclusions obtained in the aforementioned analysis of the dataset. The architecture comprises five different phases, structured in two layers, and is graphically depicted in Figure 2.

The two layers and their corresponding phases of the architecture are as follows:

- **Structure Layer**: This layer is responsible for structuring the text according to the two levels of information representation. First, the news story is divided according to the journalistic structure, and then the 5W1H elements of each part of the structure are determined.
  - *Phase 1. Journalistic Structure Segmentation*: Given as input a news item from a traditional digital media, this first module is responsible for dividing the news into the parts of the structure defined for a news item. Therefore, the output of this module is the news piece divided in `HEADLINE, SUBTITLE, LEAD, BODY` and `CONCLUSION`.
  - *Phase 2. Essential content (5W1H) Extraction*: Given as input the news piece divided in parts, this module extracts the 5W1H components from each part of the news.
- **Veracity Layer**: This layer is the crucial element of this research and its purpose is to determine the veracity of each of those parts previously detected, as well as to predict the veracity of the news piece using the veracity of the different components. Determining the veracity, as explained in Section 1 implies automatically determining the accuracy and truthfulness of a piece of information within a news document.
  - *Phase 3. Essential content (5W1H) External Enrichment*: Given the 5W1H components of the news piece, this module is in charge of enriching the information of each component by using external fact-checking knowledge.
  - *Phase 4. Essential content (5W1H) Veracity Predictor*: This module, using the annotation of all the possible features (textual and fact-checking knowledge) of the 5W1H components, classifies each component in a veracity value.
  - *Phase 5. News article Veracity Predictor*: The last module, using the veracity classification of each component, is in charge of predicting the veracity of the whole news item, which is the final output of the pipeline proposed.

The integration of the phases as a pipeline results in a prediction of the veracity of the news item. Although the Structure Layer is not the fundamental point of this research, possible approaches have been proposed for each of these phases, but without going into depth on their solution, which is a segmentation task whereas this work focuses mainly on the automatic detection of fake news. In the following sections, the development of each of the aforementioned phases is explained in more detail.

4.1. Journalistic Structure Segmentation

This phase structures the news story according to the journalistic structure previously presented in Section 2.1. Given a plain news item as input, a initial preprocessing is performed to obtain `HEADLINE` and `SUBTITLE` following a set of simple rules. These rules are variable and depend on each site's structure. After that, the remaining text will be divided into `LEAD`, `BODY` and `CONCLUSION` applying a named entity recognition approach. Using Spacy library[16], a tokenization of the news document is performed, and a set of features are obtained for each token (see Table 5). The features are defined in the Spacy library documentation[17].

Feature	**Description**
`text`	Original text of the token.
`lemma`	Lemmatized version of the token.
`pos`	Coarse part-of-speech tag, e.g., VERB, NOUN, etc..
`tags`	Several fine-grained part-of-speech tags such as person, number, tense, etc.
`dep`	Label of the token in the dependency tree.
`shape`	Syntactic representation of the token shape.
`ent_type`	General-purpose entity label, e.g., PERSON, ORG, etc.
`is_alpha`	Boolean value indicating if the token is alphanumeric.
`is_stop`	Boolean value indicating if the token is a stopword.
`index`	Relative index of the token in the document, between 0 (first token) and 1 (last token).

Table 5: Token-level features, extracted with Spacy.

News documents are segmented at the token-level using a Conditional Random Fields (CRF) model (Sutton et al., 2012) trained on the token features described in Table 5. To introduce context, each token feature set is complemented with the features of surrounding tokens (both before and

[16]https://spacy.io/
[17]https://spacy.io/api/token#attributes

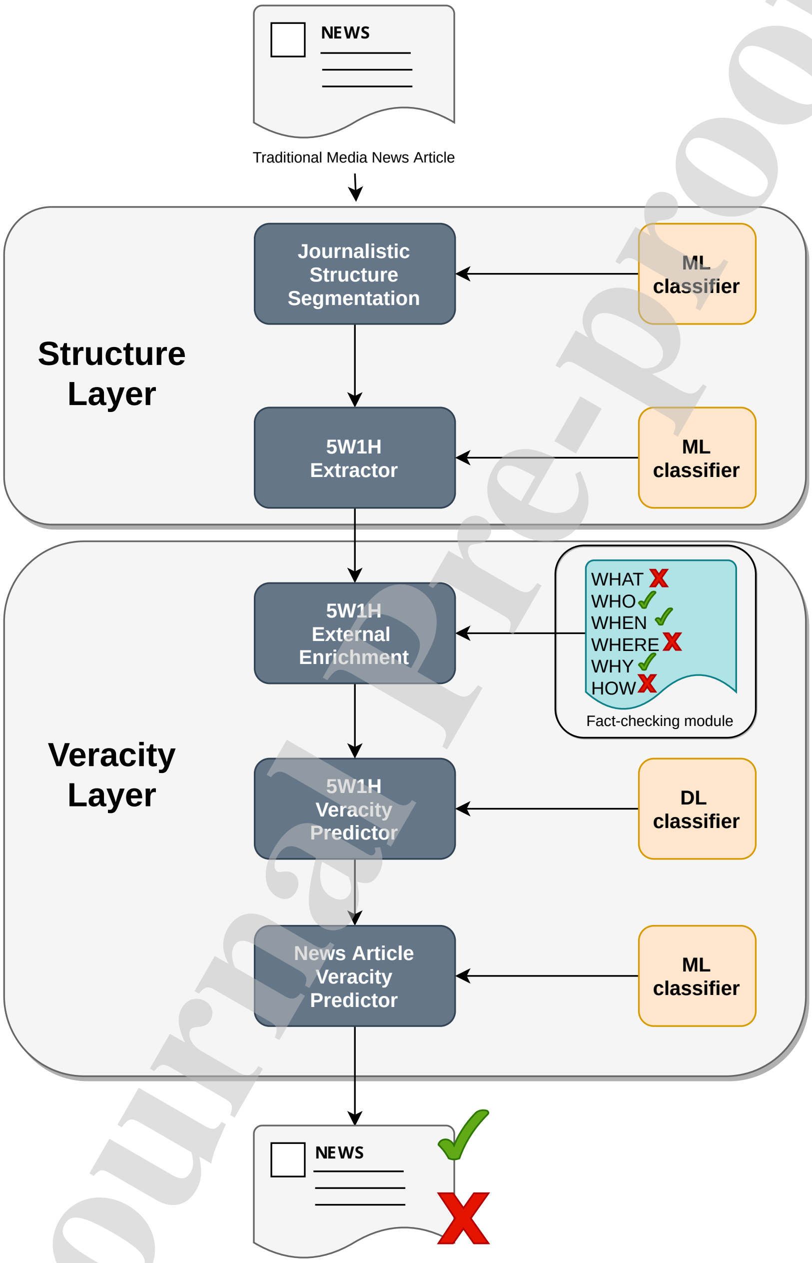


Figure 2: Pandemic Fake News Detection system's architecture

after) in a small window of size 0 to 3. This parameter can be adjusted to improve accuracy at the cost of a larger computational cost. The CRF model is trained using *sklearn-crfsuite*[18]. The segmentation problem is thus modeled as a sequence tagging problem, where each token is assigned one of these labels: `LEAD, BODY` and `CONCLUSION`.

After this process, the segmented news item is the output of this module, as shown in this example.

(2)

Token	**Features**	**Structure Part**
token1	| ... |	=> Lead
token2	| ... |	=> Lead
token3	| ... |	=> Body
token4	| ... |	=> Body
...		
tokenN	| ... |	=> Conclusion

4.2. Essential content (5W1H) Extraction

Using all the features per token previously obtained, a second CRF model is used to classify each token of each part into one of the 5W1H components, or NONE. As observed in Table 3, there is a large imbalance in the labels' distribution, which provokes a poor performance of models trained to predict all classes at once. For this reason, a two-level hierarchical classification is performed, where labels are divided into two sets: the first level consists of the most common labels (NONE and WHAT) while the least common labels are grouped in a special REST class; and, the second level comprises only the least common classes (HOW, WHEN, WHERE, WHY and WHO). This allows the training of two separate models that can deal better with the unbalanced distribution of the labels, allowing each model to only focus on a smaller set of classes for which their relative numbers are similar.

The fact that one of the features obtained by Spacy is the Named Entities (NE) is very useful in this module since they are related to some questions such as LOCATION for WHERE, PERSON/ORGANIZATION for WHO, or TIME for WHEN. Furthermore, the same features shown in Table 5 are used to represent each token. Likewise, a window size can be adjusted to include more context at the cost of a larger feature set and increased computational cost.

In the case of classes with a smaller set of examples in the dataset, in addition to the features used in the first level, the semantic roles of the text

[18]https://sklearn-crfsuite.readthedocs.io/en/latest/

will also be used in the second level of the hierarchical model. According to (Moreda et al., 2011), the use of semantic roles can improve the detection of answers to the 5W1H, especially when dealing with questions whose answer is not itself a Named Entity. For example, in this sentence where semantic roles are annotated, the role `AM_LOC` is the answer of a Where question:

(3) Where was Pythagoras born? Samos Pythagoras was born [`AM-LOC` on the island of Samos].

In order to annotate semantic roles, Freeling(Padró & Stanilovsky, 2012) is used because this tool also annotates semantic roles in Spanish.

As this module performs, the different 5W1H are detected and an example of the output obtained by this module for each news document is presented next.

(4)

Token	**Features**	**Structure Part**	**5W1H**
token1	| ... |	Lead	=> None
token2	| ... |	Lead	=> What
token3	| ... |	Body	=> What
token4	| ... |	Body	=> What
token5	| ... |	Body	=> What
token6	| ... |	Body	=> Who
...			
tokenN	| ... |	Conclusion	=> Where

As can be seen in the example, each 5W1H might span multiple tokens, as in the case of the "What" item that comprises token 3,4 and 5.

4.3. Essential content (5W1H) External Enrichment

This module is in charge of enriching each 5W1H component by using external fact-checking knowledge. As our intention is to look only for essential content, i.e. the treatment of each 5W1H element, the process is carried out using those elements rather than raw text. The first point we would like to stress is that performing a fact-checking module is not a trivial task and implies in-depth research in itself, which is beyond the scope of this work. Nevertheless, in order to add external knowledge to the proposed pipeline, a simple fact-checking module has been implemented that will be able to detect whether the 5W1H elements of a news story are part of any previous fact-check. Of course, this implies that the fact exposed in the news story has been previously refuted. The purpose of this module is not to determine the veracity of each 5W1H, but to extract external information that, in addition with the textual content, helps in the prediction of the veracity of each component performed in Phase 4.

More specifically, this module uses the Google Fact Check Tools API [19], which is based on ClaimReview markup[20]. An example of a fact-check in Spanish is shown in Figure 3.

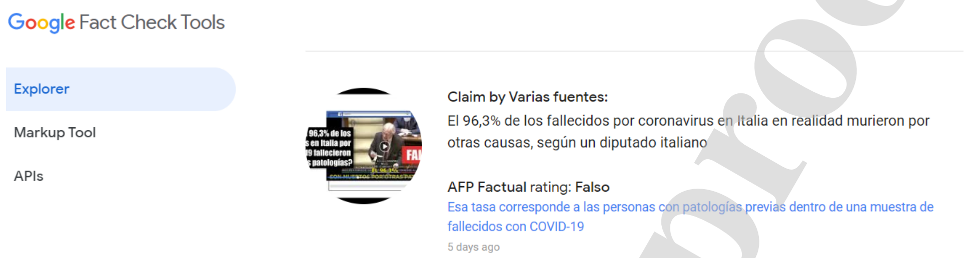


Figure 3: Screenshot of a fact-checked claim in Spanish

A JavaScript Client for the REST API[21] was implemented to access this tool[22]. The essential content of the news (5W1H) is searched as follows. For each part of the document (title, subtitle, ...) the 5W1H items are sent separately to the API for checking their veracity. If a value is found, the label is updated to that value. If any of the 5W1H items do not receive a veracity value or receive contradictory values, a second check will be done with all 5W1H items of that part to add context information. To do that, all the items will be concatenated and sent again to check their veracity. In this case, the value obtained will serve to update the veracity value of each item. The API's textual rating is mapped to one of our True/False/Unknown categories.

A simple solution is proposed for this fact-checking module, but in future the fact-checking procedure should be enhanced.

4.4. 5W1H Veracity Predictor

This phase is in charge of predicting the veracity value of each 5W1H component of each news document, based on all the evidence collected in Phase 3 plus the textual content of each element. Due to the complexity

[19] https://developers.google.com/fact-check/tools/api/

[20] https://schema.org/ClaimReview

[21] https://factchecktools.googleapis.com/v1alpha1/claims:search

[22] For further information, please consult the API documentation at: https://developers.google.com/fact-check/tools/api/reference/rest

of the task, this problem is tackled using DL, since solving the problem in this phase requires not only dealing with textual features of the components but also high level features obtained from external knowledge that enrich the components (Fact-checking in this case). In order to predict the veracity of each component, the module uses a sequential LSTM-Convolutional model with the following architecture (see Figure 4):

1. A trainable embedding layer with output dimension of 32, a maximum sequence length of 100 tokens (longer sequences are truncated) and a maximum number of 1000 vocabulary entries (built during training from the top 1000 tokens by frequency in the training set).
2. A dropout layer with a dropout rate of 0.25.
3. A 2D convolutional layer with 64 filters and kernel size of 5.
4. A max-pooling layer with a pool size of 4.
5. A second dropout layer with a dropout rate of 0.25.
6. An LSTM layer with an output dimension of 70.
7. A third dropout layer with a dropout rate of 0.25.
8. A dense layer for one-hot encoding of the label of the 5W1H component (i.e., “WHAT”, “WHERE”, “WHY”, etc.).
9. A dense layer for one-hot encoding of the label of the article part in which the 5W1H component appears in the news article (i.e., “LEAD”, “BODY”, etc.).
10. A concatenation of the previous three layers.
11. A final dense layer with 3 outputs (one for each class of *True*, *False*, *Unknown*) with a softmax activation function.

This model was adapted from a classic architecture for sequence classification proposed in the Keras ML library[23] and modified to fit the number of features and training examples available in this research. The exact parameters of each layer (e.g., layer sizes, dropout rate, number of filters, etc.) were decided after a short manual tuning among a range of sensible parameters.

When the fact-checking information is available, a parallel two-layer dense feed-forward network (with a total of 130 trainable parameters) is added, whose output is concatenated before the final dense layer with the previous model.

[23] `https://keras.io/`

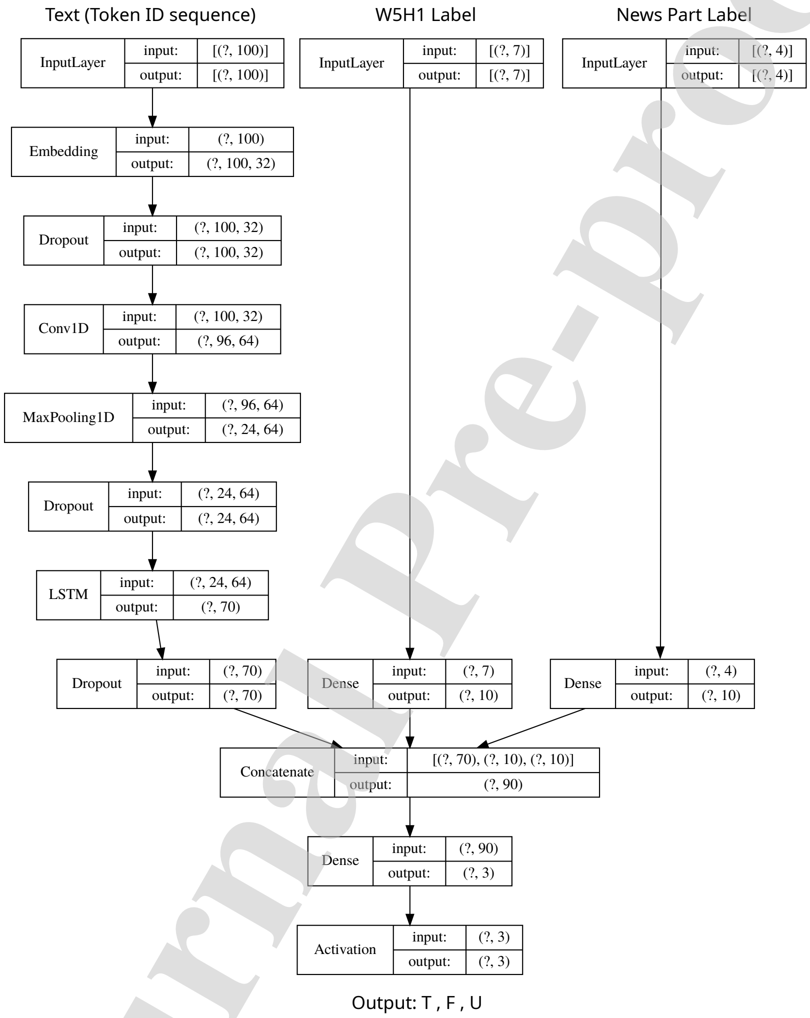


Figure 4: Graphical representation of the 5W1H Veracity Predictor DL architecture. The type of each layer and tensor shapes are reported. Shapes with size "?" indicate the batch dimension, whose size is determined at training time and does not influence the total number of parameters.

The overall model contains $80,377$ trainable parameters ($80,507$ when adding the fact-checking features), and is trained with the Adam optimization scheme using categorical cross-entropy as loss function, with the recommended hyperparameters (Kingma & Ba, 2014). To improve performance, this model is trained with early stopping, based on the loss measured on a separate 10% of the training set, with 3 epochs of patience Prechelt (1998). The model is implemented in the Python *keras* library.

The DL model is trained independently on each continuous sequence of tokens that belongs to the same 5W1H part to predict their veracity value. At the end, using all the features previously extracted in the pipeline, the module is predicting the veracity of each component.

An example of the output of this module is:

(5)

Token	**Features**	**Structure Part**	**5W1H**	**Veracity**
token1	| ... |	Lead	=> None	Null
token2	| ... |	Lead	=> What	T
token3	| ... |	Body	=> What	T
token4	| ... |	Body	=> What	T
token5	| ... |	Body	=> What	T
token6	| ... |	Body	=> Who	F
...				
tokenN	| ... |	Conclusion	=> Where	T

In this example, the "What" item of the Lead is assigned a False veracity value; the "What" item of the body is assigned a True veracity value; and the "Who" is a false element. This means that the fact explained in the body happens, however the person involved in this fact was not the one indicated in the news document. The last phase will learn that certain entities are less relevant than others, which is why we consider this phase to have a regularizing effect, like an ensemble.

4.5. News Article Veracity Predictor

Finally, the last phase is in charge of giving the final prediction of the news item, using one of several classic ML models (as implemented in the *scikit-learn* package[24]):

- Logistic Regression, with an L_2 regularization factor of 1.0 and a LBFGS optimizer.

[24] https://scikit-learn.org/stable/

- Decision Trees, using GINI as the criteria for feature selection.
- Support Vector Machines, with a Radial Basis Function kernel and a regularization factor of 1.0.
- Multinomial Naive Bayes, with a Laplace smoothing factor of 1.0.
- Random Baseline using a stratified random strategy.

In this module, to represent the documents, for each part of the structure of the document, their 5W1H items are aggregated according to their veracity value, and the number of each item within each veracity value is counted. Thus, considering that there are 5 parts in the structure (HEADLINE, SUBTITLE, LEAD, BODY and CONCLUSION), and 6 possible 5W1H types of items (WHAT, WHO, WHEN, WHERE, WHY and HOW) within each part, and each of these 5W1H items can have one of three veracity values (TRUE, FALSE, UNKNOWN) the final number of numerical features generated is 90.

<HEADLINE id=1 type='F'><WHO id=1 type='F'>Dr. Chen</WHO> affirmed that <WHAT id=1 type='F'>cancer is cured</WHAT> <HOW id=1 type='F'>by infusing water with a slice of lemon </HOW> <WHEN id=1 type='F'>every day</WHEN></HEADLINE>

<LEAD id=1 type='F'><WHAT id=2 type='T'>Lemon has several properties</WHAT>, but <WHO id=2 type='F'>medical experts</WHO> <WHAT id=3 type='F'>have used it</WHAT> <WHERE id=1 type='F'>in Asia</WHERE> <WHEN id=2 type='F'>for millions of years</WHEN> <WHY id=1 type='F'>because it cures cancer</WHY>. It is known that <WHAT id=4 type='T'>lemon has health benefits</WHAT>, but <WHO id=3 type='U'>renowned oncologists </WHO> <WHEN id=3 type='F'>now</WHEN> <WHAT id=5 type='F'>stated that it is possible to kill cancer cells</WHAT> <HOW id=2 type='F'>by consuming hot water with citrus fruits juice</HOW> <WHEN id=4 type='F'>every morning</WHEN> <WHY id=2 type='F'>since its vitamins are up to 100 times more effective than chemotherapy.</WHY></LEAD>

Figure 5: Graphical visualization of part of the annotation of a newspaper article using FNDeepML annotation scheme.

For instance, considering Figure 5 annotation of the headline and lead of a specific newspaper article, the following numerical features are extracted from headline and lead[25].

```
{
    HEADLINE_WHAT_TRUE: 0,
    HEADLINE_WHAT_FALSE: 1,
    HEADLINE_WHAT_UNKNOWN: 0,
    HEADLINE_WHO_TRUE: 0,
    HEADLINE_WHO_FALSE: 1,
    HEADLINE_WHO_UNKNOWN: 0,
    HEADLINE_WHEN_TRUE: 0,
    HEADLINE_WHEN_FALSE: 1,
    HEADLINE_WHEN_UNKNOWN: 0,
    # ...
    LEAD_WHAT_TRUE: 2,
    LEAD_WHAT_FALSE: 2,
    LEAD_WHAT_UNKNOWN: 0,
    LEAD_WHO_TRUE: 0,
    LEAD_WHO_FALSE: 1,
    LEAD_WHO_UNKNOWN: 1,
    LEAD_WHEN_TRUE: 0,
    LEAD_WHEN_FALSE: 3,
    LEAD_WHEN_UNKNOWN: 0,
    # ...
}
```

The same type of features will be generated from the other parts of the structure of the document. Each feature indicates the number of 5W1H components with a specific label and veracity that appear in each part of the news. For example, `LEAD_WHAT_TRUE: 2` indicates that the `LEAD` contains two `WHAT` items annotated with a `TRUE` veracity value. The model is trained to predict the overall document veracity label based on these numerical features.

5. Experimental Setup and Evaluation

5.1. Evaluation Measures

In order to evaluate the proposal, the commonly used NLP measures (accuracy, precision, recall and F-measure) are used.

Precision (P) is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$P = \frac{\#TruePositive}{\#TruePositive + \#FalsePositive} \tag{1}$$

[25]Only some of the features are shown to exemplify the generation of these features

Recall (R) is the ratio of correctly predicted positive observations to the all observations being actual positive.

$$R = \frac{\#TruePositive}{\#TruePositive + \#FalseNegative} \tag{2}$$

F1-Score (F_1) is the weighted average of Precision and Recall.

$\mathrm{F}_1 = 2 * \text{Precision} * \text{Recall}\frac{}{Precison+Recall}(3)$

Accuracy (Acc) is the most intuitive performance measure and it is simply a ratio of correctly predicted observations to the total observations.

$$Acc = \frac{\#TrueP + \#TrueN}{\#TrueP + \#FalseP + \#TrueN + \#FalseN} \tag{4}$$

Furthermore, the macro and micro average of each measure is given when necessary. Macro average is the average of each of the measures, whereas micro average is an average weighted by support value —which is the number of true instances for each label—. Using these measures is also important because the macro average will be poor if any class is small, but the micro average will penalise less severely in classes with very few elements. The difference between macro and micro indicates how much damage the corpus imbalance is doing to the model.

5.2. Experiments

The main objective of the experimentation proposed in this research is to demonstrate the hypothesis that because fake news is a combination of false and true information whose aim is to create confusion among readers, an adequate approach to the problem of automatic fake news detection is a two-layer architecture.

The following set of experiments for each of the layers are performed to validate our hypothesis:

- **Structure Layer performance**: A set of experiments related to the first two phases have been carried out. The proposals made are measured to assess potential areas for improvements to increase effectiveness.

  – *Phase 1 performance. Journalistic Structure Segmentation:* The performance of the module doing the segmentation into `LEAD`, `BODY` and `CONCLUSION` of the text is measured.

- *Phase 2 performance. 5W1H Extractor:* In this experiment, the performance in detecting the different segments that correspond to the answer of the 5W1H is measured.

- **Veracity Layer performance**: A set of experiments to measure the two phases that determine both the veracity of the components and the veracity of the news have been implemented. In addition, a final experiment allows the validity of this work's hypothesis to be determined by measuring the Veracity Layer as a whole. Phase 3 does not have an individual experiment since it is an enrichment phase and its validity is given by the results of phase 4, which have been measured both using the information of phase 3 and without using it to determine its benefits.

  - *Phase 4 performance. 5W1H Veracity Predictor:* This experiment measures the performance of the module that predicts the veracity value of each element of the news piece. In order to prove this and to determine the validity of this module in isolation, the 5W1H labels of the gold standard dataset have been used and the performance of the module using different configurations is measured by: i) using only the textual characteristics of the content of the 5W1H components; ii) using only the fact-checking characteristics and; iii) using the combination of both.
  - *Phase 5 performance. News Article Veracity Predictor:* To measure the accuracy of this phase in this experiment, the phase is measured in isolation, using as training the manually annotated gold standard news pieces with the different parts of the structure as well as the 5W1H elements with their veracity value. Thus, the errors of the previous phases are avoided, and the validity of this module alone is measured. This is one of the most important experiments since it proves the validity of the proposal.
  - *Phase 3+4+5 performance. Veracity Layer* This experiment aims to determine the effectiveness of the Veracity Layer but avoiding segmentation errors produced by the Structure Layer. Specifically, using the gold standard segmentation of the text, Phase 3, 4 and 5 together are performed and measured.

Finally, the performance of the full pipeline is measured and a cross-domain validation is performed to explore the applicability of our proposal across domains.

6. Results and Discussion

This section presents the results obtained in each of the experiments described in Section 5 and a discussion of those results.

6.1. Phase 1 performance. Journalistic Structure Segmentation

Table 6 presents the performance at a token level of the Structure Segmentation Module that corresponds to Phase 1 in the pipeline.

Features	*P*	*R*	F_1	*Acc*
Lead	0.851	0.772	0.810	0.851
Body	0.960	0.964	0.962	0.929
Conclusion	0.710	0.836	0.768	0.648
micro avg	0.935	0.937	0.936	0.938
macro avg	0.840	0.857	0.846	0.809

Table 6: Journalistic Structure Segmentation performance

Overall, this module obtains a micro F_1 score of 0.936 in an independent test-set of 20% of the news items. Table 7 shows the confusion matrix over the test-set. As expected from a CRF-based model, no confusion occurs between classes that never overlap, i.e., Lead and Conclusion. Since Body is the majority class (with a support of 23,708 tokens out of a total of 28,154 in the test-set, or 84.11%), it is also the class with the highest F_1. However, despite their being a significantly lower number of training instances for the remaining classes, their F_1 scores are significantly higher than what can be expected from a random baseline. By comparison, using only the token relative index produces an overall F_1 of 0.772, which is an indication that most news items (in the corpus) follow a relatively similar structure in terms of the relative sizes of each segment.

6.2. Phase 2 performance. 5W1H Extractor

Table 8 presents the performance at a token level of the 5W1H Segmentation Module that corresponds to the Phase 2 in the pipeline.

As explained in Section 4.2, a hierarchical model is trained on different subsets of classes to deal with the imbalance of labels in the dataset. As

	Lead	Body	Conclusion
Lead	2345	692	0
Body	411	22849	418
Conclusion	0	236	1203

Table 7: Confusion matrix for the Journalistic Structure Segmentation module. For each of the 28, 154 tokens in a 20% test-set, the rows indicate the real label and the columns indicate the predicted label.

a comparison baseline, a single linear model (logistic regression) trained on the complete set of labels obtains a micro-average $F_1 = 0.932$, but a macro-average $F_1 = 0.309$. This is because the model assigns a higher importance to the most common labels and hence performs very poorly on low-count labels such as WHY ($F_1 = 0.048$), HOW ($F_1 = 0$) and WHEN ($F_1 = 0.128$).

The hierarchical model is trained first only on NONE, WHAT and REST (which groups all the remaining labels), producing the results shown in Table 8(top), in a test-set of 20% of the news items. Then, a second model is trained only on the subset of tokens with labels HOW, WHY, WHEN, WHERE and WHO, producing the results shown in Table 8 (bottom) in the same test-set. The first level uses only syntactic and semantic features from Spacy, while the second step includes also the semantic role features from Freeling. This configuration showed better results, presumably because semantic roles are not useful for the recognition of the WHAT class, in contrast with the rest of the 5W1H components. As can be observed, each model is significantly better (in terms of macro F_1) in the corresponding sub-problem.

Interestingly, the first step is able to recognize the REST class exactly, which means that we can estimate the overall performance of the model by aggregating the results of both models. The combined estimated macro-average F_1 for this two-step model is 0.661, significantly higher than the 0.309 provided by a single model. Furthermore, the worst performance is obtained for the HOW and WHY labels, which have the least number of instances. If we discard these labels and only consider the remaining 5 labels (including NONE) the overall macro F_1 would be 0.774. Finally, a very good performance is obtained in the WHAT label (F_1=0.948), which corresponds to the most important element in terms of determining the veracity of a news item. The HOW and WHY elements are important in a fact-checking process to determine the veracity of a news item, since they add nuisance and detail and might thus change the deeper meaning of a news item. For this reason,

	First step		
	P	R	F_1
NONE	0.999901	0.983090	0.991425
WHAT	0.901966	0.999378	0.948177
REST	1.000000	1.000000	1.000000
macro avg	0.967289	0.994156	0.979867
micro avg	0.986880	0.985474	0.985784
	Second step		
	P	R	F_1
HOW	0.262500	0.750000	0.388889
WHEN	0.788732	0.629213	0.700000
WHERE	0.489583	0.566265	0.525140
WHY	0.336957	0.462687	0.389937
WHO	0.844444	0.639731	0.727969
macro avg	0.544443	0.609579	0.546387
micro avg	0.694253	0.611702	0.636717

Table 8: Results for the first level (top) and second level (bottom) of the hierarchical model trained for 5W1H extraction.

their failed detection in this phase is likely to cause a significant decrease in the overall performance of the pipeline. In contrast, a high accuracy in the extraction of the WHAT label might compensate for the performance loss. However, the reliable extraction of 5W1H elements in general is a difficult problem, and it is not the purpose of this research to fully address it.

The complexity of the phase 2 task is acknowledged and for this reason the literature on Automatic extraction of 5W1H presented in Section 2.1 will be taken into consideration to improve future performance of the Structure Layer.

6.3. Phase 4 performance. 5W1H Veracity Predictor

As explained in Section 5, this phase is evaluated in different configurations. Using the gold standard 5W1H elements in the dataset, the validity of this module in isolation is measured.

Table 9 presents the performance of the 5W1H Veracity Predictor Module that corresponds to the 4th phase in the pipeline, with three configurations:

Deep NN (Text) uses only textual features of the tokens within each 5W1H component annotated in the gold standard dataset.

Deep NN (FC) uses only fact checking features of the 5W1H components, automatically obtained in Phase 3.

Deep NN (Combined) uses both textual features and fact checking features of the 5W1H components.

For comparison purposes, two baselines are implemented: using a strategy that always predicts the majority class (**Dummy**) and using the TF-IDF representation of the text of each 5W1H component to train a logistic regression. The values correspond to the mean precision, recall, F_1 and accuracy of each model for each veracity label (i.e., **U**nknown, **T**rue and **F**alse), averaged across 10 independent runs with 80% training and 20% testing splits.

Models	Baseline		Deep Learning		
	Dummy	TF-IDF	Text	Fact-Check	Combined
Precision (T)	0.000	0.601	0.592	0.370	0.592
Recall (T)	0.000	0.471	0.547	0.930	0.523
F_1 (T)	0.000	0.528	0.565	0.529	0.554
Precision (F)	0.000	0.476	0.512	0.000	0.507
Recall (F)	0.000	0.234	0.374	0.000	0.452
F_1 (F)	0.000	0.313	0.424	0.000	0.468
Precision (U)	0.513	0.630	0.733	0.512	0.753
Recall (U)	1.000	0.837	0.837	0.993	0.821
F_1 (U)	0.678	0.719	0.780	0.675	0.784
Accuracy	0.513	0.607	0.658	0.542	**0.660**
Macro_F_1	0.226	0.520	0.590	0.409	**0.602**

Table 9: Performance results of different configurations of 5W1H Veracity Predictor using Gold standard 5W1H segmentation

As can be deduced from the results obtained in Table 9, determining the veracity of each of the essential contents of a news item is not a trivial task. The figures obtained make it clear that the use of textual characteristics has a limit when it comes to improving the detection of falsehood. Also, we see that combining textual information with high-level characteristics extracted from external knowledge, such as fact-checking in this case, help to improve the prediction of the veracity of each component. Obviously, the increase is limited in our case because the tools that perform with optimal results are lacking at present, and the tool we apply at present is very limited.

It should be noted that although the dataset is limited in size (200 news), this phase is trained with individual 5W1H phrases; hence, there is a larger number of training examples. In total there are 2,788 different 5W1H phrases, of which 2,230 are used for training (80%) and 558 are used for validation

(20%). With respect to features, a maximum number of $1,000$ different tokens is allowed for the embedding layer (i.e., the $1,000$ most common tokens). Similarly, a maximum of 100 tokens is allowed for any 5W1H phrase in the LSTM layer. These limits maintain a small total number of trainable parameters, which makes it feasible to achieve a better-than-baseline performance even with such a small number of training examples.

To better understand the behavior of the 5W1H Veracity Predictor in different types of 5W1H components, Table 10 shows the evaluation metrics aggregated per 5W1H label.

5W1H label		HOW	WHEN	WHERE	WHY	WHAT	WHO
micro	precision	0.564	0.626	0.581	0.642	0.519	0.991
	recall	0.495	0.558	0.535	0.571	0.505	0.994
	F_1	0.481	0.526	0.501	0.557	0.502	0.992
macro	precision	0.496	0.610	0.573	0.630	0.502	0.332
	recall	0.492	0.495	0.477	0.609	0.491	0.333
	F_1	0.442	0.485	0.457	0.573	0.485	0.332
	accuracy	0.495	0.558	0.535	0.571	0.505	0.197

Table 10: Evaluation metrics for the 5W1H Veracity Predictor model using combined syntactic and fact-checking features aggregated per type of 5W1H component.

The results obtained between the different 5W1H components are quite similar, except from WHO, that, as indicated in Table 3, has a high degree of uncertainty (U veracity), resulting in a high micro-F_1 but limited accuracy. The results indicate the need to add more complex information that implies external knowledge and context in order to improve the prediction of the veracity of each component.

6.4. Phase 5 performance. News Article Veracity Predictor

The experiments with the News Article Veracity Predictor represent the most interesting results because they demonstrate that by considering the veracity of the news structure parts and the 5W1H, a suitable solution to the problem of automatic fake news detection is provided. Therefore, to avoid problems arising from previous phases, this module is measured in isolation using the gold standard 5W1H elements and their manually-assigned veracity value. The experiment demonstrates that this information is valuable when determining the veracity of the whole news document. Table 11 presents the performance of this last phase in the pipeline. The results of the different ML approaches applied are shown as well as two baselines to determine if there is an improvement when using our proposal: i) a random baseline; and ii) a baseline using the TF-IDF of the whole document annotated with a unique veracity value for the document.

	True News			Fake News				
Model	P	R	F_1	P	R	F_1	Acc	Macro F_1
Baseline (Random)	0.523	0.503	0.510	0.483	0.502	0.489	0.502	0.500
Baseline (TF-IDF)	0.609	0.868	0.715	0.726	0.381	0.494	0.637	0.605
Decision Tree	0.971	0.976	0.972	0.976	0.965	0.969	0.971	0.971
Logistic Regression	0.964	0.997	**0.980**	0.996	0.958	**0.976**	**0.978**	**0.978**
Naive Bayes	0.920	0.995	0.956	0.995	0.902	0.945	0.951	0.950
SVM	0.934	0.994	0.962	0.993	0.919	0.953	0.958	0.958

Table 11: Results of News Article Veracity Predictor performance using veracity of gold standard 5W1H components

As can be concluded from the results in the table, all the models proposed significantly outperform the two proposed baselines. Even so, the model that obtains the best results is Logistic Regression both for detecting false news and for determining which news stories are true, obtaining a 0.978 of macro F_1. It is especially noteworthy that using the entire annotated document with a single truthfulness value (baseline TF-IDF) the macro F_1 is 0.605. These results validate the main hypothesis set for this research, i.e., that individual 5W1H components are a good predictor of overall news story truthfulness.

6.5. Phase 3+4+5 performance. Veracity Layer

To measure the whole performance of the Veracity Layer —but avoiding the errors produced by the Structure layer, i.e. segmentation modules (phase 1 and phase 2)— the gold standard elements of the dataset are used and the performance from Phase 3 to Phase 5 of the architecture is measured.

For this purpose, the 5W1H Veracity Predictor (phase 4) was run 10 independent times in different train/test splits (80%/20%), and the results of the predicted labels (in each independent test set) were concatenated. Thus, a "new" re-sampled training set is available for training and evaluating in phase 5. This allows to train the phase 5 module directly on predicted veracity labels, instead of on the gold labels, as performed in Section 6.4. Hence, if the 5W1H Veracity Predictor makes consistent mistakes on different 5W1H labels, the phase 5 module might be able to correct these mistakes in the aggregated prediction by assigning less weights to those labels. Results are provided in Table 12.

	True News			Fake News				
Model	P	R	F_1	P	R	F_1	Acc	Macro F_1
Baseline (Random)	0.551	0.549	0.548	0.498	0.500	0.497	0.526	0.522
Baseline (TF-IDF)	0.609	0.868	0.715	0.726	0.381	0.494	0.637	0.605
Decision Tree	0.736	0.752	0.741	0.724	0.696	0.706	0.726	0.723
Logistic Regression	0.842	0.783	**0.809**	0.780	0.835	**0.805**	**0.807**	**0.807**
Naive Bayes (Multinomial)	0.794	0.827	0.808	0.804	0.760	0.778	0.795	0.793
SVM	0.802	0.768	0.781	0.761	0.786	0.770	0.777	0.775

Table 12: Results of News Article Veracity Predictor performance trained and evaluated on the predicted labels from phase 4.

As can be observed, even though the results are worse than when using gold standard annotations, they are better than what could be expected if all the errors from phase 4 were carried to phase 5. Given that phase 4 at the moment obtains a maximum of 0.660 accuracy, the fact that an average 0.805 can be obtained by aggregating low-accuracy estimations for each 5W1H hints at some sort of regularizing effect. We can argue that phase 5 indeed learns to correct some of the mistakes in phase 4. This is not surprising if we consider that, in phase 5, each of the individual veracity labels for each 5W1H component in a single article can be seen as the output of a single classifier, all of which are aggregated in an ensemble fashion. Hence, even if the individual components are not very reliable (i.e., on average each 5W1H component is correct 66% of the time), the overall classifier is far more reliable. It is known that ensemble models can outperform considerably each of their components, especially when the individual components make mistakes that are mostly independent of each other (see Section 2.2). It appears that in this case, a similar effect is taking place.

6.6. Hyper-parameter search for full pipeline performance

To measure the performance of the full pipeline, we applied a hyperparameter search based on the open source library AutoGOAL (Estevez-Velarde et al., 2020). The hyperparameter search enables testing a large number of parameter values for different parts of the pipeline to find the combination that produces the highest performance. A total of 24 hours of computing resources was devoted to the parameter search, which resulted in a total of

101 different pipelines tested. The best pipeline found achieved an accuracy of 0.775 on a 5-step cross-validation with a random split of 80% of the data for training and 20% for testing. The hyper-parameter space contains several different ML algorithms for each phase as well as specific configuration parameters such as window size and optimization technique for CRF taggers (Phases 2 and 3), number of filters and size of embedding vectors in Phase 4, and whether to count 5W1H components by article part (headline, body, conclusion) or aggregated in Phase 5. The best combination of parameters is summarized in Table 13.

Phase	Parameter	Value
Phase 1	Optimizer	LBFGS
Phase 1	Window size	3
Phase 2	Optimizer	Passive-Aggressive
Phase 2	Window size	3
Phase 4	Embedding vector size	32
Phase 4	CNN Kernel size	3
Phase 4	CNN filters	103
Phase 4	CNN Pooling size	4
Phase 4	LSTM Output size	75
Phase 4	Dropout	0.1
Phase 5	Algorithm	MultinomialNB
Phase 5	Separate 5W1H in parts	False

Table 13: Best combination of parameters found for the full pipeline.

After optimization, an independent test was performed on a random selection of 40 news test sets, obtaining the results summarized in Table 14. In general, the best pipeline found obtains an F_1 score of 0.74 and an accuracy score of 0.75. It obtains a larger precision on the True class and a larger recall on the Fake class, which indicates a small bias towards classifying news as Fake.

	True News			Fake News				
Model	P	R	F_1	P	R	F_1	*Acc*	Macro F_1
Baseline (Random)	0.551	0.549	0.548	0.498	0.500	0.497	0.526	0.522
Baseline (TF-IDF)	0.609	0.868	0.715	0.726	0.381	0.494	0.637	0.605
Full pipeline	0.920	0.550	**0.790**	0.680	0.950	**0.690**	**0.750**	**0.740**

Table 14: Full pipeline performance.

6.7. Cross-domain Analysis

In order to explore the applicability of our proposal across domains, two different experiments were performed. First, a small dataset in the political domain was created and annotated according to our annotation scheme. It contains 17 fake news and 14 true news and it was used only for testing purposes. Second, in order to be able to test the system in domains other than the political one, the two Spanish corpora available in the state of the art are studied. Since the dataset (Almela et al., 2012) is not news as such, but a dataset of opinions, the FN detection system has been tested using Posadas' Spanish dataset (Posadas-Durán et al., 2019)[26], which is a dataset of news websites covering different domains (Science, Sport, Economy, Education, Entertainment, Politics, Health, Security and Society). Since this corpus is only annotated with two labels (real and fake), we can not use it for training our system, only for testing it as a cross-domain experiment. Table 15 shows the results obtained given these two cross-domain scenarios.

Training	Testing	Full Pipeline				
		Acc	F_1 (True)	F_1 (Fake)	Micro F_1	Macro F_1
Health dataset	Health dataset	0.75	0.79	0.69	0.74	0.74
Health dataset	Political dataset	0.62	0.17	0.75	0.53	0.46
Health dataset	Posadas dataset	0.52	0.31	0.59	0.43	0.45

Table 15: Cross-domain analysis of the proposal

Not surprisingly, there is a loss in accuracy and F_1 as compared to the within-domain results shown in the first row of Table 15. Similar performance losses occurred inthe literature when cross-domain is analysed (Pérez-Rosas et al., 2018) (Huang & Chen, 2020) (Hanselowski et al., 2018). Regarding Posadas dataset, there is also a considerable loss of F_1 and accuracy in comparison with results obtained by the authors (Posadas-Durán et al., 2019). One of the main causes is that Posadas' dataset comprises documents of nine different domains, whose vocabulary is very diverse and therefore dissimilar to the health vocabulary on which our system is trained. Thus, it must be considered that (Posadas-Durán et al., 2019) trained on its dataset, hence it is expected that their results are higher.

[26] Available at `https://github.com/jpposadas/FakeNewsCorpusSpanish`

The results of the cross-domain experiment show that there is still room for improving the model in addressing the cross-domain intractability issue. Although the system obtains reasonable accuracy results, the F_1 score drops significantly, especially in the True class. This can be explained by considering the imbalance in terms of features in our training set, i.e., it is harder for a news item to be classified as True, since almost any evidence of False statements points to fake news. This happens because news items with both fake and true statements are considered fake, as well as news items with only fake statements. Hence, our model inherently learns a bias towards classifying news items as fake, unless a sufficient number of True 5W1H components are present. In the extreme case of having no evidence whatsoever, our model defaults to classifying a news item as Fake. Notice that this is a sensible default, and it is not hard-coded, but learned implicitly from the annotated corpus. When applying our model out-of-domain, significantly less 5W1H components are successfully extracted, since the lexical features of the other domains differ from those where the CRF models were trained. This failure in the earlier parts of the pipeline explains the bias towards the Fake class.

7. Comparison of our proposal with the state of the art

The objective of a SOTA comparison is to make a reliable comparison. Due to the novelty and particularities of our dataset, where every essential part of the news is detected and assigned a veracity value, and since this does not occur in any other SOTA dataset, to the authors' knowledge, a direct comparison of the results of the different systems published in literature on those datasets is not possible. Nevertheless, we carried out a set of comparative experiments that compare our proposal with the state of the art in three scenarios: 1) our proposal vs state-of-the-art systems, training and testing them on our dataset; 2) our proposal's performance vs the most common method used by the SOTA approaches that use linguistic cues extracted from LIWC for detection; and 3) our proposal configured with different state-of-the-art fake news detection approaches that use ML or DL for each phase of the pipeline.

7.1. Our proposal vs SOTA systems

To make this SOTA comparison, two outstanding works in the literature (Pérez-Rosas et al., 2018) and (Rashkin et al., 2017) were analyzed. However, in both cases the systems were not available and have been replicated.

Furthermore, another outstanding work whose code was available was also included (Potthast et al., 2018).

Regarding (Pérez-Rosas et al., 2018)'s approach, and taking into account that their system is not available, we have replicated it considering the best result obtained in this research. The following features have been used as characteristics: number of characters; complex words; long words; number of syllables; word types; number of paragraphs; and readability metrics —Flesch-Kincaid, Flesch Reading Ease, Gunning Fog, and the Automatic Readability Index (ARI)—. Then, in line with how they describe their experimentation in their work, we have used a linear SVM classifier and five-fold cross-validation with our English dataset.

Regarding (Rashkin et al., 2017)'s approach, we replicated their DL model, which consists of an embedding layer (using GLOVE 100-dim pre-trained embeddings[27] which are fine-tuned during training, form (Pennington et al., 2014)), an LSTM layer with 300 hidden units, and a final dense layer. The only difference in our replication is that since our problem is binary, we apply a sigmoid activation and binary cross-entropy loss, instead of softmax and categorical cross-entropy, as in their original paper. Training parameters are also replicated, i.e., 10 epochs with a batch size of 64 items. 30 independent train/test splits were performed.

The SOTA systems used in this comparison work on English datasets where the news documents are assigned a veracity value. Therefore, to compare the performance of our proposal with other SOTA fake news detection systems, our dataset was translated into English and the three aforementioned SOTA systems were trained and tested on the translated dataset, with a train and test configuration of 80%/20% in 30 independent evaluations. The results obtained are shown in Table 16.

System	Acc	F_1 (True)	F_1 (False)	Macro-F_1
Our system	**0.75**	**0.79**	**0.69**	**0.74**
Potthast (2018)	0.66	0.63	0.69	0.66
Pérez-Rosas (2018)	0.56	0.63	0.46	0.52
Rashkin (2017)	0.53	0.46	0.55	0.51

Table 16: Comparison with SOTA systems: training and testing with our dataset

[27] https://nlp.stanford.edu/projects/glove/

As presented in Table 16, our proposal surpasses the other systems. Regarding (Potthast et al., 2018), our system obtains an improvement of 13.6% of accuracy, 25.4% in F_1 on true news and obtains a very similar result in the F_1 on false news. Regarding (Pérez-Rosas et al., 2018), our system obtains an improvement of 33.45% of accuracy, 25,40% in F_1 on true news and 50.33% in F_1 on fake news. Regarding (Rashkin et al., 2017), our system obtains an improvement of 41.51% in accuracy, 71.73% in F_1 on true news and 25.45% in F_1 on fake news.

This SOTA comparison shows that our approach improves the results obtained on our dataset, and that it is a robust solution. Furthermore, our approach is more ambitious and aims to go one step further by addressing the problem at a higher level than a simple text classification problem, whereas these systems are acting as a black box. Hence, our goal is to give the user the specific elements of the information that drives the system to a final conclusion regarding the veracity of the news article.

7.2. Our proposal vs the most common method used by SOTA approaches

Considering Section 2, most of the literature's approaches focus on studying linguistic aspects of falsehood, identifying different types of linguistic features of fake news (Zhou & Zhang, 2008) (Pérez-Rosas et al., 2018) (Almela et al., 2012) (Afroz et al., 2012) (Shu et al., 2019) (Volkova et al., 2017). Therefore, a comparison of our proposal with the method applied by many different state-of-the-art approaches was performed. According to the literature, the Linguistic Inquiry and Word Count(LIWC) (Newman et al., 2003) is widely used to extract the lexicons falling into psycholinguistic categories. Hence, our dataset was annotated using LIWC and a set of experiments were performed using different ML approaches with the final aim of comparing our proposal with one of the most common state-of-the-art fake news detection methods.

For this purpose, the library AutoGOAL (Estevez-Velarde et al., 2020) was used to search between 16 different types of ML methods (including shallow classifiers and DL approaches) for the best algorithm and its hyperparameters with respect to classification accuracy. After one hour of optimization, a total of 949 different variants of algorithms and parameters were tested. Each algorithm has different optimisable parameters, such as regularization factors, number of iterations, etc., which are not explicitly listed for space reasons. In total, 72 different parameters are optimised among all algorithms. Table 17 summarizes the results in terms of mean and standard deviation of accuracy for all of the different variants of each algorithm tested. Each iteration consists of 30 cross-validation steps with a random 80% of the news items for training and the remaining 20% for testing. The average accuracy (micro-average) among all algorithms is 0.616, which is 15.8 percent points below the best solution using our approach. Hence, by using LIWC features alone, a wide range of results can be expected, ranging from 0.18 to 0.66, depending on the specific algorithms, parameters, and training used. On average, these results do not outperform the approach presented in this research.

Algorithm	Acc (mean)	Acc (std)	Variants
NearestCentroid	0.6657	0.1441	115
MultinomialNB	0.6394	0.1228	60
ComplementNB	0.6140	0.1241	57
NuSVC	0.5822	0.2476	88
LinearSVC	0.5646	0.2580	66
Perceptron	0.4895	0.2980	76
Neural Network	0.4833	0.0236	2
DecisionTreeClassifier	0.4571	0.0356	47
ExtraTreeClassifier	0.4490	0.0723	52
RidgeClassifier	0.4345	0.2902	69
SVM	0.4267	0.2557	55
SGDClassifier	0.4151	0.2932	54
PassiveAggressiveClassifier	0.4103	0.2571	47
KNeighborsClassifier	0.3414	0.2777	56
LogisticRegression	0.3269	0.3517	49
BernoulliNB	0.1881	0.2486	56
Our approach (full pipeline)	**0.750**		

Table 17: Summary of the mean and standard deviation of accuracy for the different variants of each ML algorithm trained on the LIWC characteristics.

7.3. Our proposal using different SOTA fake news detection approaches for each step

As presented in the background (Section 2), many SOTA systems use different ML or DL approaches to solve the problem. In this case, to conduct a comparison of approaches, the AutoGOAL library (see Section 6.6) is also used to see the results of using different approaches for each of the steps in our proposal.

A total of 24 hours of computation enabled the evaluation of 101 different combinations of algorithms applied to our proposal. For each combination, we define a set of features that correspond to specific algorithms or parameters used in our pipeline. Then we aggregate for each feature the accuracy of all the pipelines in which it appeared. The same algorithm or parameter value for a given phase appears in multiple pipelines, combined with different options in the remaining phases. For this reason, the average accuracy of each feature as reported is influenced by the context, i.e., by the characteristics of the pipelines in which that feature was reported.

The total number of optimisable parameters in the pipeline is 63, ranging from numerical parameters such as the number of neurons in each layer or the dropout rate, to categorical parameters such as which algorithm is used for the last phase. We report only the most relevant parameters, i.e., those that show a larger influence in the overall performance of our proposal. Figure 6 shows a graphical representation of the most relevant parameters of each pipeline evaluated, and Table 18 summarizes the average accuracy of all the evaluated pipelines that contained the given features. The parameters reported are the following:

- Optimization algorithm used in the CRF taggers (Phases 1 and 2).
- Window size of the CRF taggers (Phases 1 and 2).
- ML algorithm used in Phase 5.
- Whether to aggregate 5W1H components by body part in Phase 5.

As can be observed, both the algorithms used in Phase 2 & 3 as well as those in Phase 5 have a significant impact on the overall accuracy of the pipelines. The most consistent algorithm for the CRF components is Passive Aggressive. For Phase 5, even if the algorithm that produces on average the best performance is Stochastic Gradient Descent, the most consistent option is Multinomial Naive Bayes, which is also the algorithm selected in the best performing pipeline (see Section 6.6). The window size in the CRF components is also a significant factor, as shown in Figure 6, since pipelines with a larger window (size=3) consistently perform better than those with a shorter window. This is an expected result since a larger window allows more tokens to be considered as part of the context for a specific token. Finally, it is interesting to note that aggregating all 5W1H components instead of counting them within the part of the article in which they appear increases the average accuracy by more than 3 percent points. This has the effect of

Phase	Feature	Value	Acc (mean)	Acc (std)	Variants
Phase 2 & 3	algorithm	Averaged Perceptron	0.5839	0.0923	28
Phase 2 & 3	algorithm	Adaptive Regularization	0.5455	0.0861	28
Phase 2 & 3	algorithm	Stochastic Gradient Descent	0.5875	0.0919	22
Phase 2 & 3	algorithm	L-BFGS	0.5921	0.0884	38
Phase 2 & 3	algorithm	Passive Aggressive	0.6072	0.0930	69
Phase 2 & 3	window size	0	0.5857	0.0859	70
Phase 2 & 3	window size	1	0.5960	0.0989	25
Phase 2 & 3	window size	2	0.5543	0.0964	29
Phase 2 & 3	window size	3	0.6061	0.0915	61
Phase 5	algorithm	BernoulliNB	0.5792	0.1145	6
Phase 5	algorithm	CategoricalNB	0.6031	0.0930	8
Phase 5	algorithm	ComplementNB	0.6500	0.1061	2
Phase 5	algorithm	DecisionTreeClassifier	0.5156	0.0566	8
Phase 5	algorithm	ExtraTreeClassifier	0.5528	0.0292	9
Phase 5	algorithm	GaussianNB	0.5350	0.0742	5
Phase 5	algorithm	KNeighborsClassifier	0.5788	0.0957	20
Phase 5	algorithm	MultinomialNB	0.6041	0.0953	37
Phase 5	algorithm	NuSVC	0.5000	0.0354	2
Phase 5	algorithm	SGDClassifier	0.7250	0.0707	2
Phase 5	algorithm	SVC	0.6000	0.0354	2
Phase 5	use parts	False	0.5978	0.0911	56
Phase 5	use parts	True	0.5661	0.0898	45

Table 18: Summary of the performance associated to the most relevant parameters of each pipeline evaluated in the AutoML process.

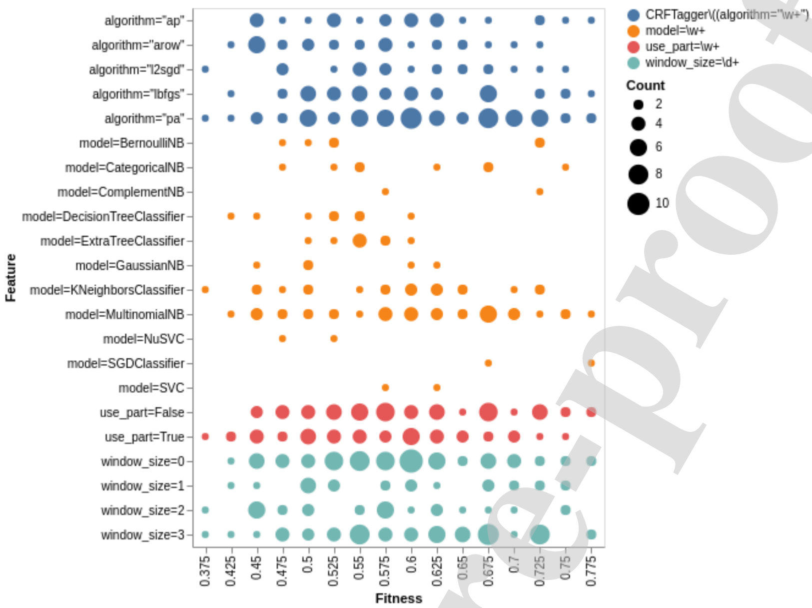


Figure 6: Graphical visualization of the most relevant parameters of each pipeline evaluated.

reducing the total number of features in Phase 5, which could help alleviate the impact of a reduced training set.

8. Conclusions and further work

This paper presents a novel approach to dealing with automatic fake news detection on traditional digital media and our proposal is based on the premise that fake news combines true and false data with the intention of confusing readers. However, to the best of the authors' knowledge, current datasets consider the news as a whole and assign it a single truthfulness value, although this truthfulness value may have degrees of certainty, but they are not determining specifically which parts within the news item are true and which parts are false or even unverifiable.

Our proposal exploits the journalistic structure of news articles and how the content is presented, following the inverted pyramid hypothesis. Moreover, the essential content of the news is typically presented by answering six questions that comprise the 5W1H. Based on this knowledge, a new fine-grained annotation scheme (FNDeepML) is defined using two levels of representation: i) Newspaper article structure and ii) Essential news content (5W1H). A new dataset in Spanish is created consisting of 200 news articles focused on the health domain, and specifically on COVID-19 news.

The proposed architecture comprises two main layers —Structure and Veracity Layers— that predict not only the article's veracity value but also that of the article's main content. The experiments have demonstrated that the use of the veracity value of the different structural elements and that of the 5W1H essential content within the news provides a suitable solution to the problem. The best performance for the Veracity Layer was obtained with a Logistic Regression model, resulting in a F_1=0.807, compared to a baseline using the TF-IDF of the entire document —annotated with a unique veracity value for the document— resulting in F_1=0.60. Furthermore, the performance of the Veracity Layer using the veracity of gold standard 5W1H components increases to F_1=0.978. These findings demonstrate the validity of our proposal.

Our experiments also demonstrate that determining the veracity of each 5W1H component using only textual information has a limited prediction performance, and therefore, adding high-level features (i.e. fact-checking information, semantic relations between components, contextual features, among others) would be beneficial. The future goal is to predict as accurately as possible the veracity of each essential element of the news, as this would be a very powerful tool for readers, who would benefit from a detailed report on the reliability of news content elements.

At this stage of the research, the news elements and the news document are classified only in True/False/Unknown categories. However, in future developments, a weighting of elements may be used to determine an overall degree of veracity depending on the Fs and Ts items detected in the same block of text. Furthermore, in future work phases 1 and 2 should be enhanced. On the other hand, phase 3 would require an improvement in the automatic fact-checking tool by means of determining the semantic relationship between the different 5W1H elements to provide a context to those items. This contribution would enable the detection of contradictions in the 5W1H relations, which may be indicative of fake information.

Acknowledgements

This research work has been partially funded by Generalitat Valenciana through project "SIIA: Tecnologias del lenguaje humano para una sociedad inclusiva, igualitaria, y accesible" with grant reference PROMETEU/2018/089, by the Spanish Government through project RTI2018-094653-B-C22: "Modelang: Modeling the behavior of digital entities by Human Language Technologies", as well as being partially supported by a grant from the Fondo Europeo de Desarrollo Regional (FEDER) and the LIVING-LANG project (RTI2018-094653-B-C21) from the Spanish Government. Furthermore, we would like to thank Difusión Comunicación, especially Tono Jordá and Lara Sánchez Belda, and Newtral for their collaboration in collection and annotation of datasets.

References

Afroz, S., Brennan, M., & Greenstadt, R. (2012). Detecting hoaxes, frauds, and deception in writing style online. In *Proceedings - 2012 IEEE Symposium on Security and Privacy, S and P 2012* Proceedings - IEEE Symposium on Security and Privacy (pp. 461–475). Institute of Electrical and Electronics Engineers Inc. doi:10.1109/SP.2012.34 33rd IEEE Symposium on Security and Privacy, S and P 2012 ; Conference date: 21-05-2012 Through 23-05-2012.

Agarwal, A., & Dixit, A. (2020). Fake news detection: An ensemble learning approach. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1178–1183).

Almela, A., Valencia-García, R., & Cantos, P. (2012). Seeing through deception: A computational approach to deceit detection in written communication. In *Proceedings of the Workshop on Computational Approaches to Deception Detection* EACL 2012 (pp. 15–22). Stroudsburg, PA, USA: Association for Computational Linguistics.

Bednarek, M., & Caple, H. (2012). *News discourse* volume 46. A&C Black.

Bonet-Jover, A., Saquete, E., Martínez-Barco, P., & Ángel García-Cumbreras, M. (2020a). Fndeepml annotation guidelines. URL: `https://doi.org/10.5281/zenodo.4091549`. doi:10.5281/zenodo.4091549.

Bonet-Jover, A., Saquete, E., Nieto, M., Belén, V. M., Piad-Morffis, A., Martinez-Barco, P., & Ángel García-Cumbreras, M. (2020b).

Fndeep dataset. URL: https://doi.org/10.5281/zenodo.4090914. doi:10.5281/zenodo.4090914.

Brennan, M. R., & Greenstadt, R. (2009). Practical attacks against authorship recognition techniques. In K. Z. Haigh, & N. Rychtyckyj (Eds.), *IAAI*. AAAI. URL: http://dblp.uni-trier.de/db/conf/iaai/iaai2009.html#BrennanG09.

Chagas, L. J. (2019). The spiral model in the text of live radio journalism. *Journal of Radio & Audio Media*, *26*, 231–246.

Chakma, K., & Das, A. (2018a). A 5w1h based annotation scheme for semantic role labeling of english tweets. *Computacion y Sistemas*, *22*, 747–755. doi:10.13053/CyS-22-3-3016.

Chakma, K., & Das, A. (2018b). A 5w1h based annotation scheme for semantic role labeling of english tweets. *Computación y Sistemas*, *22*.

Chen, Y., & Chen, H. (2014). Opinion spam detection in web forum: A real case study. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 173–183).

Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., & Flammini, A. (2015). Computational fact checking from knowledge networks. *PLOS ONE*, *10*, 1–13. doi:10.1371/journal.pone.0128193.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*, 37.

Conroy, N. J., Rubin, V. L., & Chen, Y. (2015). Automatic deception detection: Methods for finding fake news. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community* ASIST '15 (pp. 82:1–82:4). Silver Springs, MD, USA: American Society for Information Science.

Dale, R. (2017). Nlp in a post-truth world. *Natural Language Engineering*, *23*, 319–324. doi:10.1017/S1351324917000018.

Das Bhattacharjee, S., Talukder, A., & Balantrapu, B. (2017). Active learning based news veracity detection with feature weighting and deep-shallow fusion. doi:10.1109/BigData.2017.8257971.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding.

Estevez-Velarde, S., Piad-Morffis, A., Gutiérrez, Y., Montoyo, A., Munoz, R., & Almeida-Cruz, Y. (2020). Solving Heterogeneous AutoML Problems with AutoGOAL. In *Proceedings of the 7th ICML Workshop on Automated Machine Learning*. ICML.

Faustini, P., & Covoes, T. (2020). Fake news detection in multiple platforms and languages. *Expert Systems with Applications*, (p. 113503). doi:10.1016/j.eswa.2020.113503.

Ferreira, W., & Vlachos, A. (2016). Emergent: a novel data-set for stance classification. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 1163–1168). Association for Computational Linguistics. doi:10.18653/v1/N16-1138.

Gilda, S. (2017). Evaluating machine learning algorithms for fake news detection. In *2017 IEEE 15th Student Conference on Research and Development (SCOReD)* (pp. 110–115). doi:10.1109/SCORED.2017.8305411.

Gravanis, G., Vakali, A., Diamantaras, K., & Karadais, P. (2019). Behind the cues: A benchmarking study for fake news detection. *Expert Systems with Applications*, *128*. doi:10.1016/j.eswa.2019.03.036.

Hamborg, F., Breitinger, C., Schubotz, M., Lachnit, S., & Gipp, B. (2018). Extraction of main event descriptors from news articles by answering the journalistic five w and one h questions. In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries* JCDL '18 (p. 339–340). New York, NY, USA: Association for Computing Machinery. doi:10.1145/3197026.3203899.

Han, S., Lee, K., Lee, D., & Lee, G. G. (2013). Counseling dialog system with 5W1H extraction. In *Proceedings of the SIGDIAL 2013 Conference* (pp. 349–353). Metz, France: Association for Computational Linguistics.

Hanselowski, A., PVS, A., Schiller, B., Caspelherr, F., Chaudhuri, D., Meyer, C. M., & Gurevych, I. (2018). A retrospective analysis of the fake news challenge stance-detection task. In *Proceedings of the 27th International Conference on Computational Linguis-*

tics (pp. 1859–1874). Association for Computational Linguistics. URL: `https://www.aclweb.org/anthology/C18-1158`.

Huang, Y.-F., & Chen, P.-H. (2020). Fake news detection using an ensemble learning model based on self-adaptive harmony search algorithms. *Expert Systems with Applications*, *159*, 113584. doi:10.1016/j.eswa.2020.113584.

Khan, S. U. R., Islam, M. A., Aleem, M., Iqbal, M. A., & Ahmed, U. (2018). Section-based focus time estimation of news articles. *IEEE Access*, *6*, 75452–75460.

Kim, J.-D., Son, J., & Baik, D.-K. (2012). Ca 5w1h onto: Ontological context-aware model based on 5w1h. *International Journal of Distributed Sensor Networks*, *2012*. doi:10.1155/2012/247346.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. URL: `http://arxiv.org/abs/1412.6980` cite arxiv:1412.6980 Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

Lewandowsky, S., Ecker, U. K. H., Seifert, C. M., Schwarz, N., & Cook, J. (2012). Misinformation and its correction: Continued influence and successful debiasing. *Psychological Science in the Public Interest*, *13*, 106–131. doi:10.1177/1529100612451018.

Mihalcea, R., & Strapparava, C. (2009). The lie detector: explorations in the automatic recognition of deceptive language. In *Proceedings of the ACL-IJCNLP 2009 Conference* (pp. 309–312). Association for Computational Linguistics.

Monti, F., Frasca, F., Eynard, D., Mannion, D., & Bronstein, M. M. (2019). Fake news detection on social media using geometric deep learning. *CoRR*, *abs/1902.06673*. URL: `http://arxiv.org/abs/1902.06673`.

Moreda, P., Llorens, H., Saquete, E., & Palomar, M. (2011). Combining semantic information in question answering systems. *Inf. Process. Manag.*, *47*, 870–885. doi:10.1016/j.ipm.2010.03.008.

Mosteller, F., & Wallace, D. L. (1963). Inference in an authorship problem. *Journal of the American Statistical Association*, *58*, 275–309. doi:10.1080/01621459.1963.10500849.

Newman, M. L., Pennebaker, J. W., Berry, D. S., & Richards, J. M. (2003). Lying words: Predicting deception from linguistic styles. *Personality and Social Psychology Bulletin*, *29*, 665 – 675.

NIST (2011). Tac 2011 guided summarization task guidelines. URL: `https://tac.nist.gov/2011/Summarization/Guided-Summ.2011-.guidelines.html`.

Norambuena, B., Horning, M., & Mitra, T. (2020). Evaluating the inverted pyramid structure through automatic 5w1h extraction and summarization. *Computational Journalism Symposium*, . URL: `http://par.nsf.gov/biblio/10168974`.

Nyhan, B., Reifler, J., & Ubel, P. (2012). The hazards of correcting myths about health care reform. *Medical care*, *51*. doi:10.1097/MLR.0b013e318279486b.

Padró, L., & Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*. Istanbul, Turkey: ELRA.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. In *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process.* (pp. 1532–1543). doi:10.3115/v1/D14-1162.

Pérez-Rosas, V., Kleinberg, B., Lefevre, A., & Mihalcea, R. (2018). Automatic detection of fake news. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 3391–3401). Santa Fe, New Mexico, USA: Association for Computational Linguistics. URL: `https://www.aclweb.org/anthology/C18-1287`.

Posadas-Durán, J., Gomez-Adorno, H., Sidorov, G., & Escobar, J. (2019). Detection of fake news in a new corpus for the spanish language. *Journal of Intelligent and Fuzzy Systems*, *36*, 4868–4876. doi:10.3233/JIFS-179034.

Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., & Stein, B. (2018). A stylometric inquiry into hyperpartisan and fake news. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 231–240). Melbourne, Australia: Association for Computational Linguistics. URL:

https://www.aclweb.org/anthology/P18-1022. doi:10.18653/v1/P18-1022.

Prechelt, L. (1998). Early stopping-but when? In *Neural Networks: Tricks of the trade* (pp. 55–69). Springer.

Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., & Choi, Y. (2017). Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2931–2937). Copenhagen, Denmark: Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/D17-1317. doi:10.18653/v1/D17-1317.

Ren, Y., & Ji, D. (2017). Neural networks for deceptive opinion spam detection: An empirical study. *Information Sciences*, *385-386*, 213 – 224. doi:https://doi.org/10.1016/j.ins.2017.01.015.

Rubin, V., Conroy, N., Chen, Y., & Cornwell, S. (2016). Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the Second Workshop on Computational Approaches to Deception Detection* (pp. 7–17). San Diego, California: Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/W16-0802. doi:10.18653/v1/W16-0802.

Saquete, E., Tomás, D., Moreda, P., Martínez-Barco, P., & Palomar, M. (2020). Fighting post-truth using natural language processing: A review and open challenges. *Expert Syst. Appl.*, *141*.

Shu, K., Wang, S., & Liu, H. (2019). Beyond news contents: The role of social context for fake news detection. In *WSDM 2019 - Proceedings of the 12th ACM International Conference on Web Search and Data Mining* WSDM 2019 - Proceedings of the 12th ACM International Conference on Web Search and Data Mining (pp. 312–320). Association for Computing Machinery, Inc. doi:10.1145/3289600.3290994 12th ACM International Conference on Web Search and Data Mining, WSDM 2019 ; Conference date: 11-02-2019 Through 15-02-2019.

Silva, R. M., Santos, R. L. S., Almeida, T. A., & Pardo, T. A. S. (2020). Towards automatically filtering fake news in portuguese. *Expert Syst. Appl.*, *146*, 113199.

Sutton, C., McCallum, A. et al. (2012). An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, *4*, 267–373.

Thomson, E. A., White, P. R., & Kitley, P. (2008). "objectivity" and "hard news" reporting across cultures: Comparing the news report in english, french, japanese and indonesian journalism. *Journalism studies*, *9*, 212–228.

Verma, A., Mittal, V., & Dawn, S. (2019). Find: Fake information and news detections using deep learning. In *2019 Twelfth International Conference on Contemporary Computing (IC3)* (pp. 1–7).

Vlachos, A., & Riedel, S. (2014). Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science* (pp. 18–22). Baltimore, MD, USA: Association for Computational Linguistics. doi:10.3115/v1/W14-2508.

Volkova, S., Shaffer, K., Jang, J. Y., & Hodas, N. (2017). Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 647–653). Vancouver, Canada: Association for Computational Linguistics. URL: `https://www.aclweb.org/anthology/P17-2102`. doi:10.18653/v1/P17-2102.

Voorhees, E. (2001). The trec question answering track. *Nat. Lang. Eng.*, *7*, 361–378. doi:10.1017/S1351324901002789.

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. *Science*, *359*, 1146–1151. doi:10.1126/science.aap9559.

Wang, W. (2012). Chinese news event 5w1h semantic elements extraction for event ontology population. In *Proceedings of the 21st International Conference on World Wide Web* WWW '12 Companion (p. 197–202). New York, NY, USA: Association for Computing Machinery. doi:10.1145/2187980.2188008.

Wang, W., Zhao, D., Zou, L., Wang, D., & Zheng, W. (2010). Extracting 5w1h event semantic elements from chinese online news. In L. Chen,

C. Tang, J. Yang, & Y. Gao (Eds.), *Web-Age Information Management* (pp. 644–655). Berlin, Heidelberg: Springer Berlin Heidelberg.

Wang, W. Y. (2017). "liar, liar pants on fire": A new benchmark dataset for fake news detection. *CoRR*, *abs/1705.00648*.

Zhang, H., & Liu, H. (2016). Visualizing structural "inverted pyramids" in english news discourse across levels. *Text & Talk*, *36*, 89–110.

Zhou, L., & Zhang, D. (2008). Following linguistic footprints: Automatic deception detection in online communication. *Communications of the ACM*, *51*, 119–122. doi:10.1145/1378727.1389972.

- A novel Automatic Fake News detection proposal based on determining the veracity of the essential content of news articles.
- A new benchmark Spanish Fake News dataset focused on health news is presented
- A new Fake News Detection architecture comprising two layers (Structure Layer and Veracity Layer) is presented
- Each layer of the architecture involves a set of phases and each phase is thoroughly described
- Performance of each layer of the architecture is measured and analysed

Credit Author Statement

Conceptualization	Alba Bonet-Jover, Estela Saquete, Patricio Martínez-Barco
Methodology	Estela Saquete, Patricio Martínez-Barco, Alejandro Piad-Morffis, Miguel Angel García-Cumbreras
Software	Alejandro Piad-Morffis, Miguel Angel García-Cumbreras
Validation	Alejandro Piad-Morffis, Miguel Angel García-Cumbreras
Formal analysis	Alba Bonet-Jover, Estela Saquete, Patricio Martínez-Barco
Investigation	Estela Saquete, Patricio Martínez-Barco, Alejandro Piad-Morffis, Miguel Angel García-Cumbreras, Alba Bonet-Jover
Resources	Alba Bonet-Jover, Miguel Angel García-Cumbreras
Data Curation	Alba Bonet-Jover
Writing - Original Draft	Alba Bonet-Jover, Alejandro Piad-Morffis, Estela Saquete
Writing - Review & Editing	Patricio Martínez-Barco, Miguel Angel García-Cumbreras
Visualization	Estela Saquete, Alejandro Piad-Morffis
Supervision	Patricio Martínez-Barco, Estela Saquete
Project administration	Patricio Martínez-Barco, Estela Saquete
Funding acquisition	Patricio Martínez-Barco, Estela Saquete, Miguel Angel García-Cumbreras

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: