Engineering Applications of Artificial Intelligence

KD SENSO-MERGER: An architecture for Semantic Integration of heterogeneous data

--Manuscript Draft--

Manuscript Number:	EAAI-23-433R2
Article Type:	Research paper
Keywords:	Natural Language Processing Knowledge Discovery Semantic Data Integration Heterogeneous Data NERC Ontology and Knowledge Representation
Corresponding Author:	Andres Montoyo University of Alicante Department of Software and Computing Systems Alicante, Alicante SPAIN
First Author:	Yoan Gutierrez, Assistant Professor
Order of Authors:	Yoan Gutierrez, Assistant Professor
	Jose Ignacio Abreu
	Andres Montoyo
	Rafael Muñoz, Full Professor
	Suilan Estevez-Velarde, Professor
Abstract:	This paper presents KD SENSO-MERGER, a novel Knowledge Discovery (KD) architecture that is capable of semantically integrating heterogeneous data from various sources of structured and unstructured data (i.e. geolocations, demographic, socio-economic, user reviews, and com- ments). This goal drives the main design approach of the architecture. It works by building internal representations that adapt and merge knowledge across multiple domains, ensuring that the knowledge base is continuously updated. To deal with the challenge of integrating hetero- geneous data, this proposal puts forward the corresponding solutions: (i) knowledge extraction, addressed via a plugin-based architecture of knowledge sensors; (ii) data integrity, tackled by an architecture designed to deal with uncertain or noisy information; (iii) scalability, this is also supported by the plugin-based architecture as only relevant knowledge to the scenario is inte- grated by switching-off non-relevant sensors. Also, we minimize the expert knowledge required, which may pose a bottle-neck when integrating a fast-paced stream of new sources. As proof of concept, we developed a case study that deploys the architecture to integrate population census and economic data, municipal cartography, and Google Reviews to analyze the socio-economic contexts of educational institutions. The knowledge discovered enables us to answer questions that are not possible through individual sources. Thus, companies or public entities can discover patterns of behavior or relationships that would otherwise not be visible and this would allow extracting valuable information for decision-making process.
Suggested Reviewers:	Alfonso Ureña, PhD Full Professor, University of Jaen laurena@ujaen.es Expert in Natural Language Processing. Text winning, Data mining.
	Rusland Mitkov, Phd Full Professor, University of Wolverhampton R.mitkov@wlv.ac.uk He is an experto in Computational Linguistics and Language Engineering.

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:

CRediT Author Statement

Yoan Gutiérrez: Conceptualization, Formal Analysis, Writing – Original Draft

José Abreu Salas: Data Curation, Validation, Writing – Original Draft

Andrés Montoyo: Methodology, Investigation, Writing – Review & Editing

Rafael Muñoz: Conceptualization, Validation, Writing – Review & Editing

Suilan Estévez-Velarde: Software, Visualization, Writing – Original Draft

KD SENSO-MERGER: An architecture for Semantic Integration of heterogeneous data

Yoan Gutiérrez[1,a,*], José I. Abreu Salas[a], Andrés Montoyo[1,a], Rafael Muñoz[1,a], Suilan Estévez-Velarde[b]

[a] *University Institute for Computing Research, University of Alicante*
Carretera San Vicente del Raspeig s/n, 03690, Alicante (Spain)
[b] *Artificial Intelligence and Computing Systems, University of Havana*
San Lázaro y L. Edificio Felipe Poey. Plaza de la Revolución, Havana (Cuba)

Abstract

This paper presents KD SENSO-MERGER, a novel Knowledge Discovery (KD) architecture that is capable of semantically integrating heterogeneous data from various sources of structured and unstructured data (i.e. geolocations, demographic, socio-economic, user reviews, and comments). This goal drives the main design approach of the architecture. It works by building internal representations that adapt and merge knowledge across multiple domains, ensuring that the knowledge base is continuously updated. To deal with the challenge of integrating heterogeneous data, this proposal puts forward the corresponding solutions: (i) knowledge extraction, addressed via a plugin-based architecture of knowledge sensors; (ii) data integrity, tackled by an architecture designed to deal with uncertain or noisy information; (iii) scalability, this is also supported by the plugin-based architecture as only relevant knowledge to the scenario is integrated by switching-off non-relevant sensors. Also, we minimize the expert knowledge required, which may pose a bottle-neck when integrating a fast-paced stream of new sources. As proof of concept, we developed a case study that deploys the architecture to integrate population census and economic data, municipal cartography, and Google Reviews to analyze the socio-economic contexts of educational institutions. The knowledge discovered enables us to answer questions that are not possible through individual sources. Thus, companies or public entities can discover patterns of behavior or relationships that would otherwise not be visible and this would allow extracting valuable information for decision-making process.

Keywords: Heterogeneous Data, Knowledge Discovery, NERC, Natural Language Processing, Ontology and Knowledge Representation, Semantic Data Integration

1. Introduction

The current challenges of a digital society require automatic knowledge representation and discovery, which explains why it is one of the most rapidly growing fields of research in com-

*Corresponding author: ygutierrez@dlsi.ua.es (+34) 965903400 ext. 9167.
Email addresses: ygutierrez@dlsi.ua.es (Yoan Gutiérrez), ygutierrez@dlsi.ua.es (José I. Abreu Salas), montoyo@dlsi.ua.es (Andrés Montoyo), rafael@dlsi.ua.es (Rafael Muñoz), sestevez@matcom.uh.cu (Suilan Estévez-Velarde)

puter and information science. Today, misinformation causes serious problems, because decision making procedures require analysis of full data, and not partially, "data is the new oil". The information needed by organizations is scattered all over the Web in an unstructured format, making it difficult to integrate into decision making and data analysis platforms and to feed predictive models about peoples' behaviour, for example (i.e., risk detection, prevention of cyberbullying, terrorist warnings, etc.). Incorporating this knowledge into information systems requires the development of methodologies that discover relevant information and continuously update it, by enriching and integrating knowledge from various sources of structured and unstructured data. Moreover, the speed at which new information is produced poses new challenges, in terms of the continuous processing, evaluation, and refinement of the acquired knowledge.

This surplus of information - news, email, social media, blogs - greatly exceeds the human capacity for processing and consuming data and is one of the challenging byproducts of the internet's accelerated growth. Thus, building automatic systems that can extract knowledge from this flow of information has become of the most actively researched fields in Computer Science. This process, involving Big Data [1] has attracted the attention of diverse research communities, such as business intelligence, engineering, entertainment, e-commerce, and social media, among others [2, 3, 4]. At the same time, it is evident that exploiting big data is a complex problem given the decentralization and non natural integration of the data.

Research in machine learning, knowledge discovery, data mining, and more recently natural language processing has enabled the techniques and tools to handle the huge amount of information on the Internet. Tasks such as the construction of search engines [5] and recommender systems [6] are some examples. These systems could be used to improve business, health care, and policy decisions [7].

As for the different approaches relevant to knowledge discovery, a continuous spectrum of techniques can be identified, based on how much expert knowledge is used. Knowledge-based techniques use rules handcrafted by domain experts [8]. These approaches have a great degree of reliability and precision and generally allow for more complexity in the extracted knowledge. However, applying knowledge-based techniques to large amounts of data does not guarantee accurate results because the knowledge base is limited.

By contrast, the statistical approaches consist of techniques based on pattern recognition with statistical and probabilistic models [9]. They perform better with large amounts of data [10], providing better recall, but are often limited to extracting simple knowledge models, and can be more sensitive to noisy, fake, or biased information [11].

Given these mutually complementary characteristics inherent to both approaches, several hybrid approaches have been proposed. Recently, new research has emerged related to ontology learning [12], learning by reading [13] and entity embedding [14], where researchers combine techniques from machine learning, natural language processing and knowledge representation to solve more complex problems that cannot be dealt with by using only classical tools.

In addition, the design of non-monolithic learning systems, built as a set of modular components that can be combined in different ways, provides a novel approach to addressing the challenge of continuously improving learned knowledge. This challenge needs the development of computational tools that are capable of building ontologies that have automated or semi-automated processes.

This composability would allow a continuous learning system to not only improve the quality of the extracted knowledge but also to learn to tune its own internal parameters - i.e. selection of suitable technologies depending on the Natural Language Processing task - to perform better knowledge extraction in the future. It is conceivable that such a system could gradually learn

what types of basic processes -e.g., entity recognition, POS tagging, etc. - are most useful for a given domain or for a given corpus. Similarly, such a system could learn which types of probabilistic models provide the best results on a particular dataset.

Given this previous scenario, the continuous learning technologies fed by updated internet content may offer a potential solution for companies or public entities by discovering behavior patterns or relationships that would not otherwise be visible. This would allow extracting valuable information for decision-making processes. In such a scenario, standardizing data becomes a pre-requisite for effective and accurate analysis[1].

Some of the major challenges faced by organizations when integrating heterogeneous data sources include:

- **Knowledge Extraction**: a complex and time-consuming task when data sources have different formats, structures, and types. Moreover, it requires the development of coordinated automatic tools to identify relevant information for each scenario.

  This work contributes to face this challenge proposing a plugin-based architecture of knowledge sensors, that are able to process heterogeneous data.

- **Data Integrity**: Integrating data from a variety of sources can introduce contradictory and incomplete data. So, preventing this is crucial. Data quality is a primary concern in every data integration strategy. Poor data quality can be a compounding problem that can affect the entire integration cycle.

  In this sense, our proposal deals with this challenge by developing the methodology and technologies to ensure the coexistence of multiple domains of knowledge dealing with contradictory facts. Also, our solution allows the exploration and dynamic interpretation of knowledge. Finally, we applied a set of metrics to ensure the integration quality.

- **Scalability**: The phenomenon of surplus of information is addressed via knowledge extraction sensors, i.e. NERCs[15] can specialise the KD systems for specific domains. Data heterogeneity leads to the inflow of data from diverse sources into a unified system, which can ultimately lead to exponential growth in data volume [2,3].

  Our plugin-based architecture tackles this challenge because only relevant knowledge to the scenario is integrated by switching-off the non-relevant sensors. Also, we minimize the expert knowledge required, which may pose a bottle-neck when integrating a fast-paced stream of new sources.

Through the use of Natural Language Processing (NLP) technologies, we can contribute to alleviating the major difficulties and challenges of managing knowledge and integrating heterogeneous sources. NLP techniques can be used to automatically extract relevant information from unstructured data sources such as text documents, emails, social media posts, etc. Thus, this can help to standardize and integrate data from various sources, making it easier to access and analyze.

[1] https://www.dataversity.net/challenges-of-integrating-heterogeneous-data-sources/
[2] https://dzone.com/articles/3-challenges-of-integrating-heterogeneous-data-sou
[3] https://ieeexplore.ieee.org/document/8396165

The advantage of our proposal is the combination of techniques to group knowledge into the same structure from structured and unstructured data sources. We leverage techniques such as *field matching* to determine the pieces of the structured data that refers to the same concepts or relations. Also, different *information extraction* procedures may be used for unstructured data to discover relevant knowledge. Thus, our proposal is able to create a unique representation of knowledge by the integration of different heterogeneous sources such as tabular data or social networks, without the requirement of a predefined schema of data representation. In consequence, a Knowledge Graph (KG) that is built in correspondence to the data provided.

The **goal of our proposal** is to design an architecture for Knowledge Discovery that is able to semantically integrate heterogeneous data - both structured and unstructured - and build internal representations that can be adapted and integrated across multiple domains.

Compared to other proposals in the literature such as Never-Ending Learning (NELL) [16], Jin et al. [17] or Bootstrapping ontology evolution with multimedia information extraction [18], one of the main features of our proposal is its explicit handling of separate knowledge from structured and unstructured data.

This approach enables the application of different techniques based on the specific domain. Additionally, it allows for the existence of contradictions or unreliable information over time, which can be verified in the future.

The paper is organized into the following sections so as to provide a comprehensive description of our proposal: Section 2 gives a brief overview of the state of the art in the automatic extraction of knowledge from different sources of information. Section 3, describes the proposed architecture for knowledge discovery. In Section 4, we present the methodologies that can be used to evaluate KD SENSO-MERGER. In Section 5, we present a case study - i.e. combining demographic data, socio-economic data, and user reviews - where the architecture is applied. Finally, in Section 6, we present the main conclusions of the research and outline possible future lines of research.

2. State of the Art

Early works such as [19] and [20] illustrate that the inherent heterogeneity of schemes, sources, and formats for data and knowledge has driven interest in their integration for a long time.

In general, this process has to solve issues related to (i) collecting the data, (ii) extracting the knowledge, (iii) matching schemes, and (iv) merging [21]. In this section, we discuss different proposals that address the aforementioned issues. To enhance the presentation, we differentiated approaches that explicitly handle the integration of heterogeneous data 2.1 from other approaches that are able to solve one or several issues but do not handle the whole process of heterogeneous integration.

2.1. Heterogenous Data and Knowledge Integration

The *mediator architecture* together with data warehousing, are two well-known approaches to integrating heterogeneous information. Typically, in mediation setups, the data remains in its original sources, described by the so-called local schemes. Then, a global-mediated-schema is designed and mapped to local schemes, to provide the unique entry point for queries. Usually, the interface between the mediator and the sources is provided by the so-called wrappers. Two main

approaches have been studied. In global-as-view (GAV), the global schema is conceived as views over the sources. Meanwhile, local-as-view (LAV) defines the global scheme independently of the sources, with local schemes as views of the global [22]. Regardless of the case, it is necessary to define a scheme of the data. The mediator provides the user with an abstraction of the underlying models of the sources, allowing working with a well-understood interface.

An architecture with similar characteristics but not described as a mediator - is proposed in [23] to integrate legacy databases from specific domains within Boeing corporation . A knowledge-based system serves as a common data model that hosts the inference engine integrating the different databases. As the authors explained, technically skilled personnel are desirable to create and match from the global ontology to the legacy knowledge representations. Despite the authors acknowledging the convenience of supporting open information scenarios where new information sources may appear, the work did not cover knowledge extraction from sources other than the legacy databases studied.

Manually creating the schemes and mappings may pose a bottleneck to the integration process. Thus, it is desirable to automate it. In [24] authors explored the automated integration of heterogeneous XML data sources using the mediator architecture. The SAMAG system finds mappings between Document Type Definitions (DTD) based on semantic and structural criteria. They leveraged WordNet [25] to define semantic links between DTD terms through relationships such as synonymy, hyponymy, and meronymy.

The mediator architecture is also the basis of [26]. The Mediator environment for Multiple Information Sources (MOMIS) introduces the ODL_{I^3} language as a means by which to build rich representations of source schemes. They curate a Common Thesaurus to encode inter-schema knowledge. It is made of terminological and extensional relationships between classes or attribute names. Four kinds of relationships are considered. Schema-derived comes from foreign keys in relational sources schemas. Lexical-driven are based on lexical relations between classes and attribute names. Both types encode intra-schema knowledge. On the other hand, designer-supplied inter-schema relationships are directly supplied by the expert to capture domain-specific knowledge. Finally, inferred-relationships are introduced by subsumption from consistent extensional relationships validated by the expert. They also introduced the concept of affinity to measure the level of similarity between classes in different source schemes. A core contribution is the use of a clustering algorithm based on affinity to identify candidate classes for integration and as a way of dealing with semi-structured data. The process is semi-automated, requiring expert knowledge to define the ODL_{I^3} descriptions and validate relationships. The integration of unstructured data is not covered.

Defining the integration architecture as separated modules or layers has the advantage of providing modularization. By this means, roles within the process are clearly identified and defined by their interfaces allowing easy replacement of components for improved versions. This is the approach taken in [27] with their 5-layer RDF-based mediator architecture. The Source Layer acts as a wrapper of the original sources being the main requirement for the ability to export the data to XML. The XML Instance Layer handles the inputs of the XML from the previous layer. The next layer, XML2RDF, provides the bridge between XMLs and the Mediator layer. The latter is at the core of the architecture. It hosts the Conceptual Model (CM) created beforehand through ontology engineering. At the top, rests the Application Layer, consisting of different applications leveraging the unified interface provided by the architecture. It's necessary to study how to handle unstructured data or cases where we can't ensure the ability to export to XML. Moreover, as in the cases previously discussed, one potential drawback is the requirement of conceiving a conceptual model beforehand.

This is also the case of the more recent proposal described in [21]. They generalized the GaV and LaV approaches using ontologies as a conceptual schema to represent both the global view and the data sources. They distinguish four stages described next. Source Wrapping encompasses the creation of an ontology for each source to be integrated. The Schema Matching step automatically searches for mappings between the different models, making it possible for the expert to validate and modify them if necessary. Based on these mappings, a global ontology is created at the Schema Merging stage, which also runs automatically. Finally, the Query Reformulation step, a query targeted to the global view, is reformulated to queries over the local sources. This approach scales easily to new sources since the process is mostly automatic, besides the tuning of some parameters for the matching and merging procedures. However, the creation of local ontologies is a probable bottleneck in cases of fast-paced streaming of new sources.

Ontololies are also a requirement of Obi-Wan, the framework for RDF integration discussed in [28]. They also leverage the mediator architecture following the Ontology-Based Data Access (OBDA) paradigm in a Global-Local-As-View (GLAV) approach, which generalizes both GAV and LAV. Heterogeneous sources are integrated into a virtual RDF graph. It is structured as RDFS [4] ontology as well as data triplets extracted from the sources through GLAV mappings.

The cases discussed up to here do not fully address the integration of unstructured sources. Moreover, the knowledge discovery from these sources as part of the integration process needs to be studied further. In [29], the authors deal with the topic of integrating unstructured sources and KD . The iASiS Open Data Graph implements a pipeline to automatically retrieve and integrate relevant knowledge in the biomedical domain. First, they integrated parallel harvesters for the unstructured content - literature - and structured knowledge, i.e. ontologies and databases. This addresses issues related to the data collection stage commented on at the beginning of section 2. The Literature Harvester provides text content as well article-topic relations that can be derived from text or metadata. Next is the Literature Analysis stage, where new relations are discovered using different text mining tools - a kind of sensor in our proposal. This module outputs *concept-concept* derived from the text as well as *mentioned_in* relations. Meanwhile, the Structured Harvester acts as a gateway for *concept-concept* relations. Next, follows a Semantic Integration component, integrating the concept-concept relations from the Structured Harvester as well as the article-topic relations from the Literature Harvester. This stage outputs *concept-concept* and *has_mesh*[5] relations. In the end, iASiS Open Data Graph integrates the relevant knowledge. It is important to note that the process is designed to run automatically. They demonstrated the suitability of their approach in different case studies in the biomedical domain. Moreover, their architecture contemplates the automatic discovery of knowledge; however, the integration of new text mining tools, or sensors designed for other kinds of data such as geo-spatial or images are not covered.

In this section, we reviewed proposals addressing the integration of heterogeneous sources. Some interesting conclusions can be drawn from this analysis. First, the mediator seems to be the preferred approach for integration because it reduces redundancy, is easier to maintain, and provides real-time or near-real-time integration in the case of fully automatic approaches. It also tends to scale better in fast-changing environments. However, the necessity for further research in this area is underscored by the the need to minimize the requirement of expert knowledge to build global and local ontologies. Moreover, it is convenient to fully automate the whole process.

[4] https://www.w3.org/wiki/RDFS

[5] Medical Subject Headings (MESH) https://www.nlm.nih.gov/mesh/meshhome.html

Also, it is key to deal with the scarcity of approaches that consider a knowledge extraction stage, or indeed the lack of studies that focus on open domains

Our proposal addresses these issues. First, we aim to fully automatize the knowledge extraction challenge with a plugin architecture that enables the easy incorporation of knowledge extraction tools - sensors - into the system. Sensors can be designed for sources varying from text, images, or geographic data, thereby maximizing the degree of heterogeneity the system is able to handle. The sensor architecture also contributes to the scalability challenge. Irrelevant knowledge can be avoided by simply switching off the respective sensors. One of the main contributions is that our proposal tries to minimize the requirement of expert knowledge - besides the creation of the sensors. The ontology holding all the knowledge is created while the data is integrated into a fully automated dependency injection mechanism. This makes the approach independent of the domain by design. Of course, the nature and domains of applications will depend on the available sensors. A final comment is that we did not conceive our architecture as a mediator as previously described . It also has elements of the data-warehousing approach, such as in [29], where relevant knowledge for the scenario is extracted and stored in a unified knowledge base.

The rest of the section is devoted to discussing approaches that are related to knowledge representation and reasoning (section 2.2), the use of machine learning to extract knowledge (section 2.3) and the KD process (section 2.4). These areas are highly relevant to our proposal as they address the key issues that need to be solved to integrate information from heterogenous sources.

2.2. *Knowledge Representation and Reasoning*

The problem of discovering, storing, and using knowledge in a computationally effective way has been extensively studied [16, 30, 12, 17]. This issue has been addressed from two different but complementary research areas: the fields of knowledge representation (Section 2.2) and machine learning (Section 2.3). The knowledge representation community provides the means to computationally represent and manage stored knowledge properly. Conversely, the machine learning community provides tools for deriving useful knowledge from large collections of structured and unstructured data. Also, there is a third field -Knowledge Discovery (Section 2.4)- which involves both knowledge representation and machine learning.

Since the dawn of computer science, one of the problems that have attracted wide attention is that of representing knowledge in a computational format, such that automatic reasoning can be performed to discover new, previously unknown truths [31].

Arguably, the most popular knowledge representation technology is the use of ontologies [32] that have become the *de facto* standard. Ontologies can be defined as a formal specification of a conceptualization [33]. This represents concepts, relations between these concepts, instances of these concepts, and inference rules for deriving new relations.

As such, ontologies can be considered as a combination of two predominant approaches for knowledge representation: those based on formal logic [34]; and those based on graphs of semantic relations [35].

The primary objective of constructing these knowledge structures is to facilitate the exploration, comprehension, and reuse of information by both humans and machines across various applications. These applications include answering questions, locating pertinent content, comprehending social structures, and making scientific breakthroughs. However, the considerable scale and intricacy of these knowledge graphs pose a significant challenge, especially when it comes to mining information across diverse topic areas Aggarwal et al. [36].

2.3. Machine Learning to Extract Knowledge

The field of machine learning provides tools for the automatic extraction of information and knowledge from different sources of data. This field not only permits the automation of processes and tasks associated with knowledge discovery or text mining but also provides a large improvement in the scalability of these processes [2].

Supervised and unsupervised learning are the two most prevalent approaches in machine learning [9]. Supervised learning can be used for recognizing specific elements of knowledge in a data source. For example, tagging pieces of text to indicate that they define an entity [37] (e.g., a person, organization, or place), recognizing relations between said entities, or assigning a sentiment or opinion score [38] to a fragment of text. On the other hand, unsupervised learning can help identify relevant structures in a large set of data. Clustering algorithms can be used to detect similar concepts or to extract abstract concepts from groups of more concrete elements. Other techniques can be used for reducing the amount of information, for example, to remove noisy, uncertain, or irrelevant pieces of information [39].

In recent years, there has been an increased interest in the problem of automatically learning relevant representations. Word embeddings [40] and more general entity embeddings [14] represent the first step towards powering deep learning approaches with more explainable internal representations. Ontologies are, by definition, representations of a given conceptualization. Therefore, using them —or similar semantic representations as information seeds of a given domain–will enhance the performance of data mining processes based on machine learning

2.4. Knowledge Discovery

Recently, a new discipline –ontology learning– has emerged, which draws ideas and techniques from both the knowledge representation and the machine learning fields. This discipline deals with the problem of automatically building ontological representations of knowledge from a variety of data sources. As such, the theory of ontology learning is relevant to the design of this proposal. Ontology learning searches to automate part of the process of creating and maintaining ontologies. This discipline has the potential of reducing the cost of creating and, most importantly, of maintaining large and complex ontologies [12]. As a sub-discipline, Learning by reading [13]is a field that draws techniques from natural language processing and knowledge representation and reasoning research areas. The purpose is to build a formal representation of fields given unrestricted related textual data. This representation must also allow fully automatic reasoning.

In ontology learning, two general high-level tasks can be distinguished: ontology population [41] and ontology enrichment [18].

Ontology population deals with the sub-problem of finding new instances for an already defined ontology, while ontology enrichment deals with adding new concepts and relations to an existing ontology. There is an overlap between these tasks, and most of the existing approaches cannot be classified purely in these terms. In this field, several tools have been proposed, that combine different approaches and solve different subsets of ontology learning tasks. Some of these systems are listed as follows:

- SYNDIKATE [42]: early approach to populating a knowledge base, with a predefined ontological structure (classes and relations).
- ARTEQUAKT [43], SOBA [44] and, WEBtoKB [45]: extracts knowledge and exploits the semi-structured format of web resources.

- VIKEF [46]: Extracts knowledge from structured data.
- ADAPTATIVA [47]: provides a bootstrapping strategy, where human experts give feedback about the extracted knowledge.
- OPTIMA [48] and ISODLE [49], OntoLT [50]: describes basic NLP techniques to extract knowledge from text.
- LEILA [51], Text2Onto [52], KnowItAll [53]: provides NLP techniques based on statistical models.
- OntoGain [54], ASIUM [55]: extracts entities and relations from text, and builds a hierarchy of concepts.
- BOEMIE [56]: infers abstract concepts from instances, on text, images, and videos.
- NELL [16]: extracts knowledge continuously from a stream of web data.
- CogKGE [17]: builds knowledge graph embedding.

In general, these tools are focused on the extraction of knowledge and on the task of discovering relevant knowledge. When extracting knowledge from a trustworthy source, even a natural language source, it makes sense to focus on optimizing recall, i.e., obtaining as much information as possible. When the input source consists of medical papers or the main web page of an institution, there is a high probability that most of the information present in those documents is correct. Hence, an ontology extraction procedure that maximizes recall will obtain good results.

However, when the input source is of lesser quality, such as blogs or social media posts, there is a greater likelihood that some, or even most, of the information is fake or incorrect. In this case, the information needs to undergo a deep semantic analysis to establish the writer's stance, i.e. positive, negative, etc.

In this context, the problem of extracting useful knowledge from a large internet-based corpus is a problem of filtering and selecting relevant information.

Despite the existence of some general-purpose systems, so far no proposal has been identified that can learn from the multitude of sources of information by activating relevant and specialized extraction and mapping sensors (i.e., Named Entity Recognition, Geolocation mapping, Sentiment Analysis, Text Categorization, aspect-based semantic analysis, and many others) that will help to determine the intrinsic characteristics of the analyzed data.

Another challenge is to obtain a computational representation of this knowledge, independent of the domain, source, and format.

In previous works [57, 58, 59, 60, 61, 62] authors explored different NLP tasks related to the proposal, such as semantic knowledge discovery, semantic enrichment, Automatic Machine Learning (AutoML), creation of a knowledge base to detect emotions and others related ones. In this work, we leverage these advances to solve the aforementioned challenges. We propose a novelty architecture for the semantic integration of heterogeneous data. Also, we present a case study to evaluate our approach.

3. Methodology

In this section, we present KD SENSO-MERGE, an architecture for the semantic merging of knowledge from heterogeneous data.

This architecture deals with the main issues described in Section 2, attempting not only to recognize information but also to reduce irrelevant information and obtain information with more relevance. Additionally, we design an architecture where new knowledge is discovered and incorporated into the existing knowledge base.

This process occurs in a feedback cycle, where new sources of information are processed and the extracted knowledge is merged with the existing knowledge. For this purpose, the framework requires three main modules, as shown in Figure 1.

As shown in Figure 1, the methodology proposed takes as starting point a document, which can be structured (1) or unstructured (1'), and depending on its typology, the corresponding module (Structured Data Processing or Unstructured Data Processing modules) is activated to convert it into an internal format that enables its processing by the Generation module. This Generation module applies different techniques to identify and extract information and generates a temporal semantic structure from the document (2). The Evaluation module takes as input this structure (3.1) and the previous knowledge existing in the system (3.2). The evaluation module uses field matching and information extraction techniques to link knowledge portions by adding non-existing connections. These new connections, depending on their quality degree [6], are returned to the generation module (3.3) to create the final knowledge base (4.1). Once, the Generation module produce this knowledge will be used as Previous Knowledge (4.2) for the next documents, and so on.

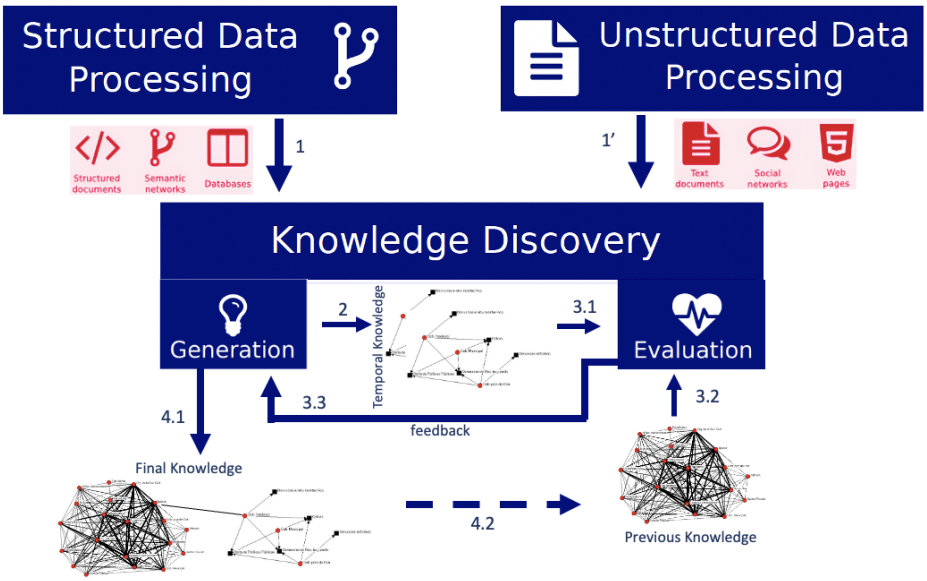


Figure 1: Simplified architecture of the framework and workflow.

The modules responsible for processing structured and unstructured data are designed to extract knowledge from these specific sources. In both cases, the output is an ontology that

[6] precision or other quality measure

represents the temporal knowledge acquired. This knowledge is not stored yet but sent to the module for knowledge discovery, where further processing is applied.

In addition to these main modules, three more modules (i.e. Algorithms, Long Term Memory(LTM); Organizational Ontology) are considered, which are interrelated. Each module within the architecture has a specific responsibility to define the inputs and outputs, enabling effective intercommunication with the other modules. Figure 2 shows a general overview of the architecture.

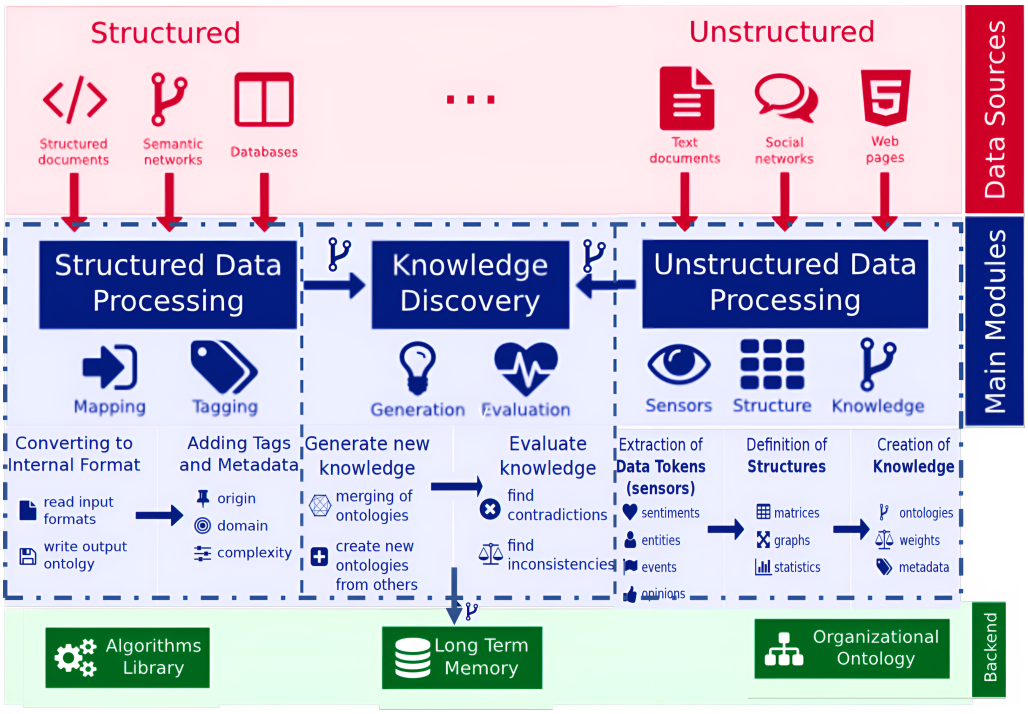


Figure 2: Overview of the architecture of the learning knowledge. Examples of the sources of information are represented at the top. The three main modules (Section 3.1, 3.2, 3.3) are highlighted at the center, with the most important processes that occur in each. The utility modules appear at the bottom.

The use of ontologies provides to this architecture effective skills for representing knowledge in a wide variety of domains and scenarios [63]. They are flexible enough to adapt to a particular domain and powerful enough to represent complex concepts. However, one of the most complex tasks in this sense is maintaining an ontology up-to-date with respect to the massive amount of unstructured data that is generated and published every day. Therefore, the need arises for computational tools to build ontologies with automated or semi-automated processes. Ontologies present a structure that facilitates the integration of new entities and their relationships. Furthermore, they are able to expand from previously created knowledge bases. The use of ontologies benefits the expansion of previously generated knowledge and its adaption to specific use cases.

Our architecture has some advantages over those presented in the state of the art. For example, it does not depend on a predefined database schema. Also, it is not necessary the supervision of human experts for giving feedback about the extracted knowledge. Our system treats het-

erogeneous documents as structured or unstructured in the same architecture, enabling specific sensors to be used depending on the problem to be tackled.

In Figure 2, the top layer (Data Sources) illustrates the input data sources for the architecture. The middle layer consists of the primary modules responsible for processing the input data and extracting valuable knowledge embedded within. Additionally, Figure 2 shows the sub-processes taking place within each module. These main modules maintain communication with one another by sharing ontologies, which represent the intermediate states of the acquired knowledge. This continual flow through the architecture ensures constant improvement.

The inner workings of the main modules are elaborated in the following Sections: 3.1, 3.2, and 3.3. The bottom layer, referred to as the Backend, comprises modules that are utilized by the rest of the architecture. These include a library of common algorithms and a centralized storage service:

Algorithm Library: Different algorithms or mathematical models are contained in the Algorithm Library for solving specific problems. This library contains metadata about the algorithms which are analyzed by the architecture, which selects the right tool for a given task. For each algorithm, its performance can be measured according to the task involved, see the evaluation alternatives in Section 4. In the near future, this architecture will automatically manage, measure and evaluate modules, algorithms, and parts, if the knowledge of how these parts interact is described in an organizational ontology inside the architecture. The technology inside it is AutoGOAL [58].

Long Term Memory (LTM): It serves as the repository for all the knowledge accumulated by the other modules, functioning as a storage repository within the architecture. Each stored ontology within the repository contains metadata that provides a description of its content.

Organizational Ontology: It holds an ontological representation of the architecture, allowing it to utilize knowledge of its own behavior in all internal processes. The details of this ontology can change according to specific implementations of the architecture's functionality. A convenient implementation of this ontology, in a computational solution, could be the base of a fully automated dependency injection mechanism, which would govern what modules can interact with each other.

3.1. SDPm: Structured Data Processing module

This module is dedicated to processing structured data, which can be found online in various formats. When it comes to representing information, there are different types of structures available, such as relational databases, concept maps, and knowledge graphs. However, we propose the use of ontologies due to their semantic richness. Ontologies have been chosen over other formats like DTO (Data Transfer Object) because they offer greater expressiveness. Given the abundance of diverse structured formats, the initial stage of this module involves converting any of these representations into a standardized internal format, specifically an ontology. This conversion is achieved through a mapping process, ensuring uniformity for internal use [64, 65, 66].

The problem of mapping the content of relational databases and other structured data sources into ontologies has received considerable attention since the inception of the Semantic Web [67]. Several semi-automatic and fully automatic approaches exist, with varying degrees of flexibility. An interesting example is the Relational.OWL framework [68], which performs fully automatic conversion from a relational database to an OWL schema. On the other hand, other tools such

as MAPONTO [65] require the user to provide an initial seed of correspondences before a full mapping can be produced. Using a combination of the techniques with machine learning that power these systems can provide a solution to the mapping process. The second stage of this module is to add metadata tags to the imported sources so that they can be easily integrated later with similar sources.

Formally, this stage can be defined as:

$$\begin{aligned} mapping : o &\rightarrow o' \text{ where:} \\ o &= < C, R > \text{ is a is an ontology (where } C \text{ are the space of concept and } R \text{ are relations),} \\ o' &= < C', R' > \text{ is an ontology in the internal format.} \end{aligned}$$

A key characteristic of structured data sources is that they almost always include a manual refinement or evaluation by domain experts. Hence, structured data sources tend to include mostly relevant and reliable information. The overall pipeline of this module can be conceptualized as a traditional Extract, Transform, and Load (ETL) process [69, 62], where structured data is extracted from a variety of formats, transformed to a common representation (in the form of an ontology) and loaded into the architecture for further analysis.

Afterward, the normalized and tagged block of knowledge, stored as an ontology, is passed on to the knowledge processing module for further refinement and storage purposes. To illustrate, Section 5 shows the resulting knowledge graph obtained by our architecture when fed with four datasets (three structured and one semi-structured) related to social and demographic information on educational institutions .

Formally, this stage can be defined as:

$$\begin{aligned} tagging : o &\rightarrow < o, M > \text{ where:} \\ o &= < C, R > \text{ is an ontology,} \\ M &= < (m_1, v_1), \ldots, (m_k, v_k) > \text{ is a set of metadata labels } (m_i) \text{ and values } (v_i). \end{aligned}$$

3.2. UDPm: Unstructured Data Processing module

Unstructured data sources come in a wide range of formats and computer representations. While text is a prevalent form for storing and conveying human knowledge, other forms of communication, such as images, sound files, and videos, are also valuable and gaining popularity.

In contrast to structured sources, unstructured sources exhibit significant variation in terms of reliability and completeness. For example, within natural language corpora, sources such as books and encyclopedias tend to be more relevant, self-consistent, and complete than online sources such as social media, blogs, and other comments. However, the latter is broader and more dynamic than the former. A well-known case is the problem of Wikipedia reliability, where different articles may have different quality and reliability [70], and even seemingly well-written articles may temporarily contain errors or false information [71].

Furthermore, unlike structured sources, unstructured data lacks a predefined structure of concepts and relationships. Depending on the required analysis, it is possible to extract events [72], entities [73], facts [74], sentiments and opinions [61], actions [60] and many other types of elements. Considering the diverse range of elements involved, the unstructured data processing module is structured as a pipeline. This pipeline facilitates the processing and transformation of simple concepts into more complex ones.

The unstructured data module at Figure 2 shows three stages to identify relevant information.

The first stage is called the "sensory level". Within this level, there exists a series of processing units referred to as "sensors." These sensors are responsible for extracting various pieces of data. For example, one type of sensor might extract entities from a natural language corpus supported by NER (Named Entity Recognition) technologies [73]. Another sensor could extract sentiments or opinions using Sentiment Analysis tools [61], while another could perform POS tagging [75], recognize events [72], detect actions [76], etc. A particularly useful sensor for textual information might rely on the construction of semantic representations of text (e.g., using Abstract Meaning Representation trees [77]), which can then be used to identify many of the semantic modules of a given sentence, such as Subject-Verb-Object (SVO) structures [57].

In general, each sensor carries out a specific analysis and generates a stream of *data tokens* of a particular type. These data tokens represent individual units of semantic information, such as the presence of a specific entity or the association between an entity and an event. Importantly, these data tokens are not interrelated with each other.

This is similar to the way humans obtain independent chunks of information from the environment using the senses. At the sensory level, it is possible to add new forms of extracted information and include different sensors. This allows the architecture to be expanded to accommodate additional sensors as needs arise.

Formally, this module can be defined as a set of three functions, one for each stage:

$$\begin{aligned} \textit{sensorial} : T &\rightarrow A \text{ where:} \\ T &= < t_1, \ldots, t_n > \text{ is a set of documents (e.g., in natural text),} \\ A &= < a_1, \ldots, a_n > \text{ is a set of semantic annotations (e.g. entities extracted from the text).} \end{aligned}$$

The second stage is called the"Structural Level". During this stage, the extracted data tokens from the original source are collectively processed to uncover an underlying structure. For example, starting from entities extracted from a corpus of natural language, this stage would build a graph interrelating them, by connecting those entities that appear together (e.g., in the same sentence) in the original corpus. Some of the techniques that might be used in this process come from the signal analysis domain, such as Latent Semantic Analysis (LSA) [78], Principal Components Analysis (PCA) [79], Word Embeddings [80] and many clustering techniques. Typically, the output of this stage is a representation of the underlying structure of the previously extracted data tokens, which can take the form of a graph, a correlation matrix, or a statistical description. This stage can be considered as an automated rendition of the knowledge construction processes described in the METHONTOLOGY method [81]. To continue with the human analogy, we can consider the information extracted so far as a form of "temporal memory". As such, it is noisy and full of spurious or irrelevant information, but it contains all the relevant pieces of knowledge within.

Formally, this stage can be defined as:

$$\begin{aligned} \textit{structure} : T \times A &\rightarrow G \text{ where} \\ G &= < A, E > \text{ is a graph like-structure,} \\ E &\subseteq \{e_i \in A \times A\} \text{ is a set of relations between annotations.} \end{aligned}$$

The third and final stage is called "Knowledge Level". During this stage, the structured information that was previously constructed undergoes analysis to refine and eliminate noise, while extracting the pertinent pieces of knowledge. This process facilitates the synthesis of the accumulated knowledge by considering the contextual relationships between the semantic

units extracted in the previous stage. Some of the types of analyses that can be performed at this stage include: extracting relevant information through clustering or noise filtering; inferring rules on the underlying structure of the data (i.e. using A-Priori); and, building predictive or generative models. Several tools for machine learning and artificial intelligence can be used for these analyses, such as Neural Networks, Bayesian Classifiers, Logistic Regression, and Hierarchical Clustering [82].

The output of this stage is a list of triplets, which is subsequently transferred to the Knowledge Discovery module (refer to Section 3.3) for further integration with the stored knowledge. The resulting ontology then becomes a part of the stored knowledge within the architecture. This stored knowledge is subject to iterative refinement, correction, and enhancement, as new knowledge is continually extracted from diverse sources. This iterative process is similar to "long-term memory" in humans in the sense that only the most relevant information is remembered over time. From this perspective, the architecture simulates this process so that elements that appear consistently and possess a sufficient degree of certainty are stored. In this process, the Ontology Population and Ontology Enrichment tools are used. An example of this process can be found in Section 5.

Formally, this stage can be defined as:

$$\begin{aligned} knowledge : T \times A \times G &\rightarrow < o, M > \text{ where} \\ o &= < C, R > \text{ is a classic ontology } (C \text{ are the classes and } R \text{ are the relations}), \\ M &= < (m_1, v_1), \ldots, (m_k, v_k) > \text{ is a set of metadata labels } (m_i) \text{ and values } (v_i). \end{aligned}$$

3.3. KDm: Knowledge Discovery module

The knowledge discovery module receives the output from both unstructured data processing and structured data processing, which are always in the form of ontologies. Each ontology represents a collection of knowledge assets either from a specific domain or a general domain. It is possible for some entities to overlap, meaning they contain the same knowledge facts, despite being labeled as different entities or relationships. There might be instances where contradictions or inconsistencies exist within individual ontologies or between different ontologies, as depicted in the knowledge discovery module in Figure reffig:architecture Since the architecture learns incrementally, a new ontology created in one of the previous modules can be checked against the relevant part of the previously stored knowledge. To accomplish this objective, the knowledge discovery module undertakes two primary tasks: generating new knowledge and evaluating its validity.

The knowledge generation process in this module involves two main processes: ontology merging [83, 66] and the creation of new ontologies (or more general-domain ontologies) derived from existing ones [84, 85]. Ontology merging requires the module to perform entity, relation, and instance matching between two or more ontologies that are considered similar based on specific metadata values, such as the domain [86]. After the merger, a new ontology is created that combines the matched entities and instances from the source ontologies into a single item. As a hypothetical example, a database about pharmaceutical products and their uses could be processed and stored. Then, as a new source of unstructured data emerged, providing a set of research papers that contain information about several diseases, these new sources would be processed and a new ontology would be obtained. This new ontology would relate to the diseases and treatments dealt with by the research papers [87]. By mapping the entities in this hypothetical

ontology with the existing knowledge about pharmaceutical products [88], a richer ontology is created. Another more complex process is the creation of new knowledge (new entities or new relations) based on the inference of general rules extracted from the bulk of knowledge available.

Formally, knowledge generation can be defined as:

$$
\begin{aligned}
\textit{generation} : o' \times M \times \Theta \quad &\rightarrow \quad < o^*, M > \text{where:} \\
o' \quad &= \quad < C, R > \text{is an ontology,} \\
M \quad &= \quad < (m_1, v_1), \ldots, (m_k, v_k) > \text{is the metadata,} \\
\Theta \quad &= \quad < \theta_1, \ldots, \theta_m > \text{is the subset of ontologies considered relevant for } o'. \\
o^* \quad &= \quad < C^*, R^* > \text{is the newly created ontology.}
\end{aligned}
$$

Given the incremental nature of the knowledge that is stored in the architecture, inconsistencies, and contradictions are expected to exist between the different ontologies stored. The evaluation process eventually filters and refines the inconsistent or contradictory knowledge, see Section 4. In this hypothetical example, when two or more ontologies whose domain is new to the architecture are evaluated, with a medium level of reliability, and are stored, they may have contradictory facts. In this case, because the architecture is not confident enough of its knowledge of this domain, these new ontologies would be evaluated with a medium level of reliability and stored. In time, as new ontologies of the same or similar domain are incorporated, the existing knowledge is re-evaluated, and its reliability increases or decreases accordingly. In this way, the architecture is never complete and can be self-aware of the quality of the knowledge that underpins the evolving framework.

Formally, the knowledge evaluation can be defined as:

$$
\begin{aligned}
\textit{evaluation} : o' \times M \times \Theta \quad &\rightarrow \quad M_T \text{ where:} \\
M_T \quad &= \quad < m_1, \ldots, m_k > \text{is a set of evaluation metrics.}
\end{aligned}
$$

4. Quality Metrics

In this section, we present a methodology to evaluate KD SENSO-MERGE and obtain metrics that can validate its performance in the wide variety of tasks that the architecture should be capable of conducting.

Software engineering metrics are suitable for evaluating the quality of software systems. These systems should be highly modular and extensible so that they can be easily adapted to new input formats, or new algorithms can be easily plugged in and integrated throughout the pipeline. Modular architecture design can help achieve a high degree of extensibility.

In addition to the previously mentioned high-level metrics, each of the tasks performed by the architecture can be evaluated separately. Most of these tasks have a defined performance metric that can be used to rate the degree of correctness of that task. For many of the tasks described in the previous sections, we can find standard performance metrics in the literature that can be used to evaluate each process.

Each of the different tasks performed by the architecture can have a very different baseline performance. A 90% precision can be a very good result in some complex tasks, such as dependency parsing [89], but mediocre for other tasks, such as image classification [90]. Moreover, this baseline number can vary not only across tasks but in the same task, according to which test suite (or corpus) is used.

There are also several evaluation metrics and methodologies available for the general problem of ontology learning, such as OntoRand [91] and OntoMetric [92].

Some of the most commonly described approaches in the literature for evaluating ontologies [18] are: **M1 -Comparison with a gold standard**, where a learned ontology is compared with a baseline ontology for the same domain [93]. **M2 - Expert evaluation**, where an alternative middle ground to the previous approach is having a domain expert (or several) to simply look at the resulting ontology and evaluate it according to some predefined metrics [30]. **M3 - Evaluation through an application**, where a more practical approach consists of finding an interesting application and evaluating if the use of a learned ontology provides an improvement in the application [94]. **M4 - Data-driven evaluation**, where a data-driven evaluation can be performed, by comparing the entities and relations in an ontology to a corpus of data that is not used during the construction of the ontology but is representative of the same domain [95].

Evaluating a single ontology learning method is a complex task, as demonstrated by the multiple approaches proposed by the research community. Therefore, it is highly unlikely that we can find a single automated metric to measure the overall performance of our architecture. The best approach - adopted by us - is to use a combination of existing methods, adapted to the knowledge discovery scenario, with the added complexity of dealing with multiple ontologies at the same time. In some cases, a gold standard can be found and used to obtain a benchmark comparison. In other cases, provided that a suitable interface is added to easily query the knowledge, a domain expert can interact with the architecture and give a qualitative assessment of the domain of interest. From a pragmatic perspective, the most interesting and valuable evaluation is that which finds relevant practical problems that can be solved or improved.

5. Case Study: Social and Demographic analysis of educational institutions

In this section, we present a case study designed as a proof-of-concept of our contributions. We aim to illustrate how the proposed architecture handles the semantic integration of heterogeneous data such as text, geospatial and tabular. We choose a domain that corresponds to a requirement of the project "Study of the technological needs to generate a Geo Artificial Intelligence system in public administration"[7], developed by the Center of Digital Intelligence[8], Alicante, highlighting the potential of the KD SENSO-MERGE architecture to leverage publicly available data. We analyzed the socio-demographic context of educational institutions (EI) by merging data from the different sources described in Table 1.

However, it is worth noting that the quality of the integrated knowledge will depend on factors such as the type of data, its availability, and the performance of the sensors. Also, other domains may be studied as long as it allows for demonstrating the integration of heterogeneous sources.

We consider educational institutions officially recognized in Spain and, for the sake of simplicity, only up to the high school level. An EI has attributes such as name, latitude, longitude, type (whether public, private, or mixed), phone number, etc. We also gathered data from the census sections (CS), which are the smaller territorial administrative units for statistical data collection and management within a municipality in Spain. The CS has a delimited cartography that allows operations such as the location verification of a place (e.g. an educational institution). Also, the CS has attributes linking statistical data such as age distribution, average income, etc. In addition, user reviews about EIs are considered an unstructured source.

The next section describes the realization of the main elements of the architecture to prepare and process the data 5.1 as well as the knowledge generation stage 5.2.

[7] https://cenid.es/proyectos/geointeligencia-artificial-podcast/
[8] https://cenid.es

Description	Type	Source
Census Section Cartography. Specifies the shapes delimiting the CS. Includes the province, sector area, and other location data but no statistical information about the CS. **name:** *cs_geo*	structured	Spanish Government Open Data Portal [9]
Census demographic and economic data. Provides different indices for the CS such as total population, or disaggregated by gender, place of birth, and average income among others. It does not contain any cartographic data. **name:** *cs_income*, *cs_population*	structured	Spanish Statistics Institute [10]
Dataset of educational institutions. Includes the name, location, whether it is public, private, or mixed, as well as other identification data. It does not contain data about the CS in which the EI is located or opinions about the EI. **name:** *ei_data*	structured	Valencian Community Geospatial Data Catalog [11]
Google Maps Reviews. Google user opinions about educational institutions. This dataset has a structure in the sense that it contains well-defined fields such as EI identifier, user identifier, as well review text. However, the later field is plain text, thus unstructured. **name:** *ei_reviews*	semi-structured	Google Maps [12]

Table 1: Data sources used for the case study. Within Section 5 we will use the name as a fancy identifier to easily refer to the dataset.

5.1. Case Study: Data Processing

As shown in Table 1, we considered several datasets. Here, we discuss their details as well as the core responsibilities and results of the data preprocessing components of the architecture.

From the *cs_geo* dataset we obtained the following fields:

- the CS official identifier (*cusec*). A 10-character string encoding the municipality, district, and section of the CS.
- the cartography (*geometry* of the CS. A polygon is given by a set of geographic coordinates.

The other datasets related to CS are *cs_income*, and *cs_population*. Average income by CS, as well as total population and inhabitants by place of birth. The relevant fields are:

- section identifier (*section*). This dataset uses a different encoding for the CS identifier since it comes as a string together with the municipality name. We need to preprocess this field to remove the unnecessary data to enable the mapping to the *cusec* field from *cs_geo*.
- the average income (*average_income*) of people living within a CS.

- the total population *population_total* of the CS, as well the number of people born in Spain *population_spain* and abroad *population_abroad*.

The source *ei_data* includes detailed information for an EI. We retain the following data:

- the official name (*name*) of the EI. A string of arbitrary length.
- the levels of education (*level*) covered by the EI program, e.g. secondary, high school, etc.
- the (*regime*) of the EI. A nominal field indicating whether the EI is public, private, or hybrid.
- geographic coordinates *latitude* and *longitude*

Finally, the *ei_reviews* dataset collects user reviews from Google Maps related to the EI. It includes:

- the (*name*) of the EI. It depends on the API used but it is possible to search reviews using the *name* from *ei_data* and the target region, e.g. Alicante, Valencian Community, Spain. Otherwise, we may need to map this field to *name* in EI to improve the coverage of the discovered knowledge.
- the screen name (*author*) of the user who wrote the review.
- the (*score*), a numeric value from 1 to 5 representing the score given by the *author* of the review.
- the content (*text*) of the review.

The next section discusses how the knowledge discovery loop proposed by KD SENSO-MERGE is carried out.

5.2. Case Study: Knowledge Discovery

This section describes the process of Knowledge Discovery from the datasets in Section 5.1 and the sensors used.

5.2.1. Sensors

For the case study, we used two different sensors, one designed to extract information from plain text and the other from structured geographical data. In this section, we cover their main aspects.

The dataset *ei_reviews* contains reviews as well as the score the reviewer assigned to the EI being analyzed. With the appropriate sensors, KD SENSO-MERGE can provide very valuable information not explicitly given in the dataset. In our case, we leverage a *topic classification sensor* to find out if the review is talking about one or more of the following topics within the domain:

- *staff*: captures opinions related to the professors, management, service staff as well other relevant stakeholders within the EI.
- *education*: the comment is about the values, goals, curricula, educational methodology of the EI, etc.

- *infrastructure*: refers to the conditions of the building, sports areas, and gardens of the EI among others.
- *environment*: the review talks about the surrounding neighborhood of the EI, parking facilities, parks, people, etc.

We used a text categorizer[13] designed for the NLP library spaCy[14]. We tuned the model using 10 to 20 examples of sentences encompassing the topics, extracted from *ei_reviews*. When analyzing the text, the model assigns a score from 0 to 1 representing the likelihood of the topic being covered by the text. If the *score* ≥ 0.3 we considered the topic is included.

The other sensor handles location data, i.e. latitude and longitude, as well cartography data. Commonly, certain entities such as the CS, neighborhoods, or countries have associated information about their geographical limits or perimeters. Other EIs have their location as an attribute. The *located in sensor* can detect this kind of information, creating a new relation between entities, in our case, indicating the CS where the EI is located. The dataset *cs_geo* is provided in shapefile[15]. We use GeoPandas[16] to handle this data. This library has functions to manage the perimeter of the CS as well as to verify whether an EI location is within it. However, other libraries/approaches can be used to implement this sensor.

5.2.2. Steps involved in applying the process

We started loading the four mentioned datasets into the architecture whereby all the data were transferred to the Un/Structured Data Processing modules (UDPm and SDPm, respectively) depending on the type of data involved. The architecture can infer the data types automatically or from the user's column labeling, notifying the respective modules for appropriate data management before transferring the temporal memory (i.e. triplets extracted) to the Knowledge Discovery module (KDm). Noticeably, when a field of the dataset contains string data, NLP sensors can be activated by the Unstructured Data Processing module to detect entities, events, or any category to be discovered or matched with semantic pieces already existing in the Long Term Memory(LTM). In the case of a field containing categories, it is labeled as *:rel* to treat it as concepts to be matched.

5.2.3. Loading dataset one: cs_geo

First, we loaded *cs_geo* dataset generating temporary triplet-based structures. For this dataset, the triplets had the following format: subject-predicate-object. Here are some examples: *cusec(value)-municipality-municipality(value)*; *cusec(value)-district-district(value)*; and so on. For this dataset, the triplet is focused on linking the first column (*cusec*) as an identifier of entities to the rest of the attributes. Figure 3 shows a snippet of this data in tabular format.

Since all the data is structured, the SDPm takes care of this first step. This process to generate triplets is applied to both modules UDPm and SDPm.

Next, the output of the SDPm, a list of triplets, is transferred to the KDm to be stored (considering a merging process if necessary). Due to no knowledge having been recorded in advance,

[13] https://spacy.io/universe/project/classyclassification
[14] https://spacy.io
[15] https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf
[16] https://geopandas.org/en/stable/

section	geometry:attr
3014.01001	POLYGON[(-0.4783011103641627 3, 38.34417517039852), (-0.4783027334266349, 38.34417158767038), (-0.4783171416358447, 38.34414525411608)]
3014.01008	POLYGON[(-0.49036680459935167, 38.34615206413387), (-0.4903536056030057, 38.34611764483372), (-0.4903398128258657, 38.346082893849605)]
3014.01017	POLYGON[(-0.484327156033468, 38.34639956829743), (-0.48424292828128623, 38.346106273605564), (-0.4841660275010249, 38.34597489534194)]

Figure 3: Snippet of the cs_geo dataset in tabular format.

no conflicts or merging issues were handled by the KDm to store these triplets in the LTM (i.e. neo4j for example).

At this point, we can query our knowledge base (KB) only to get information about the CS, i.e. their *cusec* or *geometry*. Figure 7a shows a view of the KB after this step and contains only instances of the CS entity.

5.2.4. *Loading dataset two:* cs_income *and* cs_population

We proceeded to load, and depending on the type of data, UDPm or SDPm is activated. We tagged all fields, but the *section* as attributes to avoid the discovery of other entities. Figure 4 shows an example of the data.

section	average_income:att	population_total:att	population_spain:att	population_abroad:att
3014.01001	17759	1146	915	231
3014.01008	21884	774	709	65
3014.01017	18767	721	552	169

Figure 4: Snippet of the dataset containing income and population in tabular format.

In this case, the architecture, through the KDm, recognizes information that refers to the same CS, producing the merging process that updates the graph. This stage does not introduce changes to the structure of the KB, but CS instances have additional attributes now.

5.2.5. *Loading dataset three:* ei_data

SDPm deals also with the structured data of the *ei_data* dataset, an excerpt of the data is shown in Figure 5. However, when the KDm receives the list of triplets generated from the *ei_data*, it activates the *located in sensor*, which is capable of finding the area, i.e. CS, to which another entity belongs. This enables the detection of location-related fields as well as areas given by fields like *geometry*. Next, the appropriate relations between CS and EI are created. Other fields from *ei_data* are included as attributes for EI entities.

institution	level:attr	regime:att	longitude:att	latitude:att	section:re
IES JAIME II	INSTITUTO DE EDUCACIÓN SECUNDA	Público	-0.470363	38.362228	03014.03037
CEIP EUSEBIO SEMPERE	COLEGIO DE EDUCACIÓN INFANTIL Y	Público	-0.503569	38.370572	03014.06020
CEIP MANJÓN-CERVANTES	COLEGIO DE EDUCACIÓN INFANTIL Y	Público	-0.482196	38.35745	03014.03006

Figure 5: Snippet of the ie_data dataset in tabular format. Content in Spanish Language.

Up to this point, the module used for this process is the SDPm since no unstructured data is involved. The resulting list of triplets that encodes the pieces of knowledge is reviewed by

checking for consistencies or missing data, and there were none in our case study. Once the checks are done, the list of triplets is stored in the LTM via the KDm. Also, this module is in charge of the merging process, i.e. updating attributes and relations of existing entities. New entities are stored and linked to existing ones. Figure 7b illustrates a view of the KB after loading the data. As it can be observed, it contains new instances of the EI entity as well the relations to the EI introduced by the *located in sensor*. See Figure 7a for a previous state of the KB.

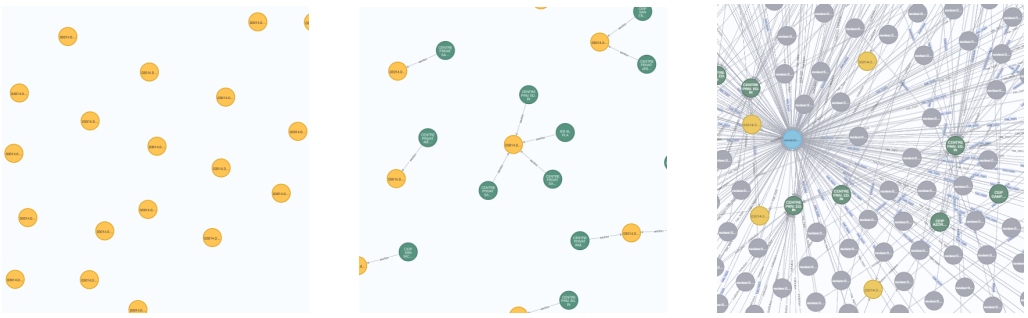
5.2.6. *Loading dataset four:* ei_reviews

Finally, ei_reviews were fed, and the fields *name*, *author*, and *score* were processed by the SDPm while the field *text* was handled by the UDPm. Then, the KDm maps the field *name* to its match *ei_data* and is leveraged by the architecture for linking the *reviews*, seen as an attribute, to the corresponding EI. Figure 6 shows a fragment of this data in tabular format. It is worth noting that the dataset didn't include the topics covered within the topic.

comment	institution:rel	author:attr	rating:att	text:attr
review:00025	CEIP MANJÓN-CERVANTES	anónimo	5	Buenos profes y buenas instalaciones. Mis hijos van muy contentos al colegio.
review:00478	CEIP SAN GABRIEL	anónimo	5	Muy buena escuela. Con una línea educativa compleja y firme.
review:00002	IES JAIME II	anónimo	5	Centro educativo muy completo con profesores muy profesionales.

Figure 6: Snippet of the ie_reviews dataset in tabular format. Content in Spanish Language.

The field *text* is analyzed at the sensory level. As a proof of concept, we activated the *topic classification sensor* presented in Section 5.2.1. This is useful to get a deeper understanding of not only how people scored the EI but also the most relevant topics for them. As a result, a list of triplets provided by both modules is merged into the LTM, ensuring the coexistence and improvement of the information at each loading round. Figure 7.7c illustrated the structure of the KB after this step. Instances of the CS, EI, and reviews are presented as well as the relations between them.



(a) KB after loading *cs_geo* and *cs_stats* (b) KB after loading *ei_data* (c) KB after loading *ei_reviews*

Figure 7: View of the structure of the Knowledge Base after each dataset is loaded. The yellow nodes represent CS instances, the green nodes EI instances, the grey ones the reviews, and the cyan the topics.

After these steps, KD SENSO-MERGE managed to build a KB populated with instances from the entities CS, EI, Reviews, and Topic, as well as the semantic relations between them. Figure 8 shows the concepts and relations discovered.

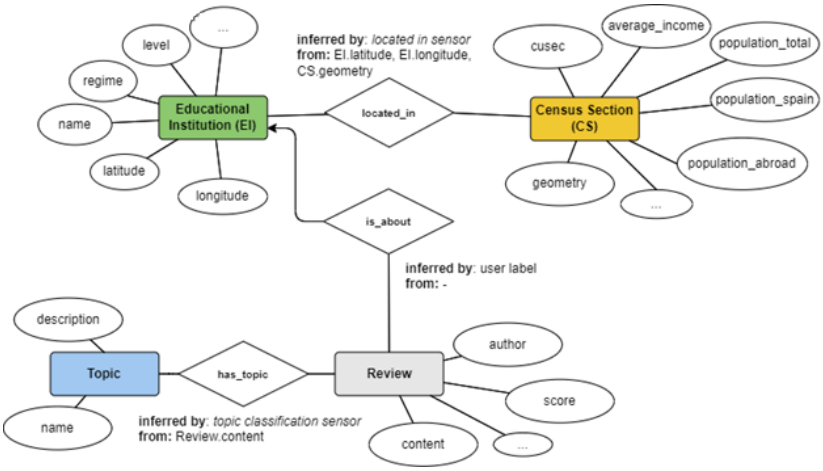


Figure 8: Concepts and relations discovered by KD SENSO-MERGE, Near relations, a note about how they were inferred.

5.3. Case Study: Knowledge Exploitation

After KD SENSO-MERGE has ingested the datasets, we obtained the distribution shown in Table 2 which collects the counts of instances in each dataset and identifies the number of discovered relations.

Instances are the basic piece of data, for example, in ei_data it comprises a record with all that is registered for an EI. Relations is the number of relations discovered from the dataset, and KB Size is the number of relations in the KB.

Dataset	Instances	KB Nodes	KB Relations
cs_geo	243	243	0
cs_stats	243	243	0
ei_data	133	376	133
ei_review	1811	1948	4213
TOTAL	2430	2810	4346

Table 2: Resume of the number of instances and relations created after each dataset is analyzed.

Automatically merging data —geospatial, entity, and topic linking– and discovering knowledge enables questions to be answered that would not be possible through the individual sources. For example, we would want to know ”the CS with average income below 10.000 euros, with at least one public EI that has reviews with a score below 3 and that talks about the staff ”. Figure 9 shows a subset of the KB answering this query.

Using the geographic data, we may also analyze visually the localization of the EI together with the average income. Figure 10 shows this information for the city of Alicante, in Spain. This is a view of a Geographic Information System (GIS) that was developed leveraging the knowledge from KD SENSO-MERGE which saved us from the tiresome work of merging the data, modeling the entities, and creating the relations.

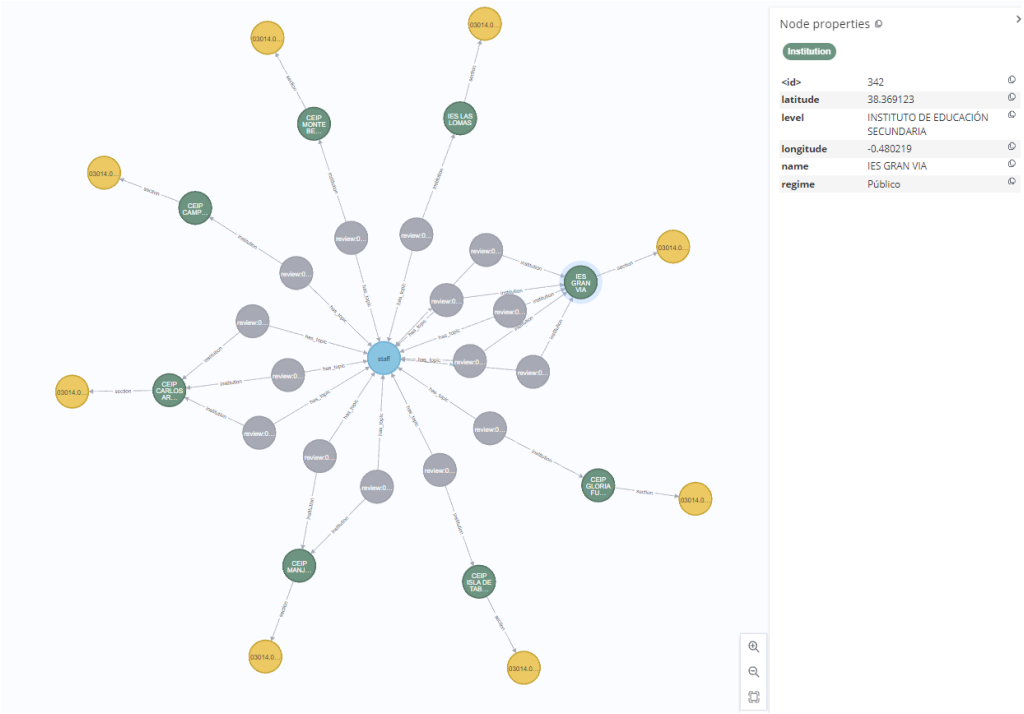


Figure 9: Subset of the KB showing the CS with average income below 10.000 euros, with at least one public EI that has reviews with a score below 3 and that talks about the staff. Results from the integration of *cs_geo*, *cs_stats*, *ei_data* and *ei_reviews*.

There are different approaches that face challenges similar to our case study. For example [96] faces superficial area challenges by applying machine learning for surface water quality prediction; and [97] deals with the fluctuations of groundwater level which are modeled by employing ensemble deep learning techniques. The two cited researches have in common with our proposal that they handle heterogeneous data (i.e. numbers and geolocations) to make predictions. However, for both of them, geolocation is only a descriptive data, and therefore not integrated into the predictive model. On the contrary, in our use case, the machine learning (i.e. regardless of soft or deep) is applied to enrich the data for further semantic and statistical analysis, but all data is integrated into the KB.

5.4. Case Study: Evaluation

The dataset was transformed into triplets to represent entities, attributes, and relations with other entities. Many of these entities were automatically recognized and generated as new concepts; others were linked to each other via geolocalization or topic. All of these procedures were evaluated by the architecture to ensure quality knowledge would be stored in the Long Term Memory (LTM).

As was mentioned in Section 4 we have different evaluation metrics. The user should pick the most suitable metric in each case. Here, we analyze the cases of M1 - Comparison with a gold standard and M2 - Expert Evaluation.

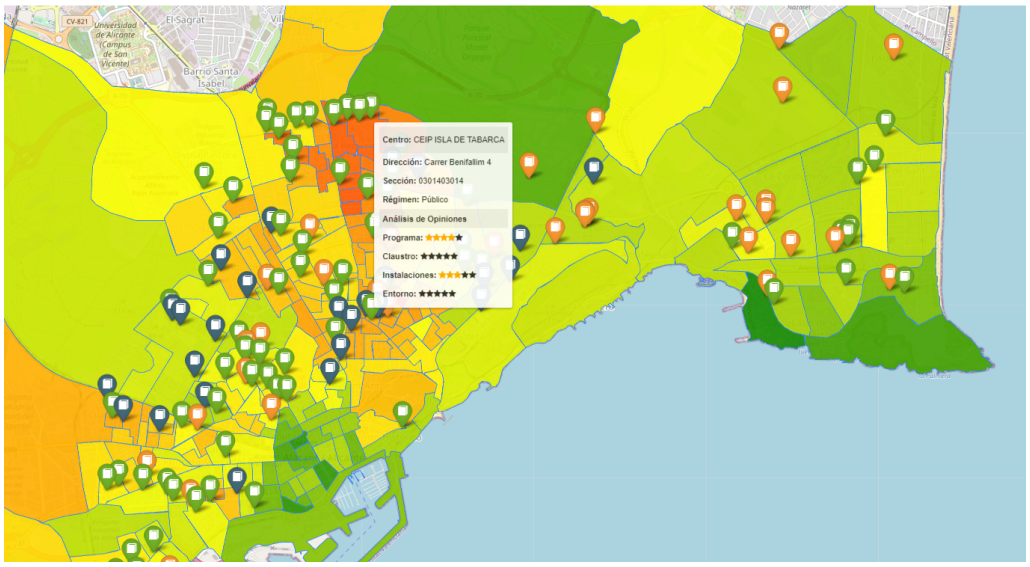


Figure 10: Map showing the CS where the EIs are located. Public EI is represented with the green marker, blue for private, and blue for hybrid. For the CS background, green represents higher incomes. Labels are in Spanish since we use a dataset in this language.

In our case, relations produced by sensors are the source of errors. For other relations, the map can be perfectly inferred given the datasets shared common features that are used for linking between datasets. For the *located_in* relation, we can determine with precision where each EI is located. Thus, we can use M2 to check if the relations were inferred as expected. In this case, 100% of the expected 133 relations were created correctly.

The *has_topic* relation is created by the *topic classification sensor*. We used M1, however, a gold standard needed to be prepared beforehand. We randomly picked 100 reviews and annotated the topics. Table 3 shows the specific quality metrics, as part of M1 (See Section 4) applied to this Case Study. It is worth noting that modeling a state-of-the-art topic classifier is beyond the scope of this work whose main purpose is to illustrate the role of the sensory level. As expected, the results are discrete in terms of precision and recall, but suitable to illustrate the procedure users can follow to evaluate their deployment of KD SENSO-MERGE. Work on optimizing the sensor, choosing another approach, or optimizing hyper-parameters may lead to notably better results.

Topic	**Precision**	**Recall**
staff	0.68	0.69
education	0.22	0.24
infrastructure	0.21	0.20
environment	0.02	0.02

Table 3: Results of M1 evaluation respect to the relation *has_topic*.

Recall and precision can be translated into an estimation of the quality. Figure 11 shows different effects of these metrics on the extracted knowledge. The recall accounts for the capability of the system to identify the entities and relations in the content. Low recall could mean we will miss some entities or relations, such as M_1 in 11. In turn, low precision implies the introduction of spurious elements, such as W_1.

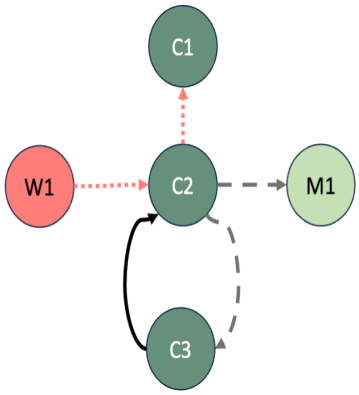


Figure 11: Different cases for nodes and relations. C_i correct nodes, M_1 missing entity, W_1 and spurious entity discovered by a sensor. The black arrow represents a correct relation, grey dashed missing ones, and, red dotted incorrect relations.

6. Discussion and Conclusions

This research resulted in the design of KD SENSO-MERGE, a knowledge discovery modular architecture that semantically integrates heterogeneous data, both structured and unstructured. The architecture builds internal representations that are adaptable across multiple domains. As the modules perform specific tasks, they communicate with each other and continuously merge different knowledge fragments.

We designed a case study that shows the capabilities of the KD SENSO-MERGE. It represents a particular knowledge discovery scenario for which the architecture provided a solution. The case study illustrates each of the main modules of the architecture, as well as their interconnections.

This work identifies three challenges which have been resolved with promising results . Regarding **Data Integrity Challenge**, KD SENSO-MERGE is capable of dealing with conflicting and contradictory facts when performing inference. Unlike most ontology learning systems, which assume that the final ontology obtained is correct, our architecture explicitly models a degree of reliability for each extracted knowledge fragment, using different evaluation metrics. The architecture was designed from the outset to deal with uncertain or noisy information. Over time, older knowledge is reevaluated, reinforced, or discarded, as new facts are discovered. This makes the architecture dynamic, in the sense that stored knowledge is always evolving, mirroring actual knowledge in human brains. In future, we plan to manage the ontology data as digital entities, so as to create profiles and model their behaviour over the time.

The **Knowledge Extraction Challenge** has been addressed through the evaluation of the case study, which corroborated that sensor performance is fundamental for the quality of integrated

knowledge. This is of particular importance for NLP sensors or for machine learning-based sensors in general. The values obtained from our case study — about 0.2 recall and precision for some topics — may render a sensor useless for most applications. Nevertheless, we did not aim to develop a state-of-the-art topic detector as the main goal was to demonstrate the sensor's role within the architecture. However, in a real-life scenario, users need to pay attention to metrics such as recall and precision since they are indicative of missing or spurious knowledge. To tackle this, we plan in future to implement AutoML [58] strategies to automatically manage, measure, and evaluate modules, and algorithms.

Finally, the **Data Scalability challenge** has been dealt with in the case study. We can conclude that developing new knowledge extraction sensors is mandatory for opening up new possibilities to create new knowledge. This enhances the deploying of KD SENSO-MERGE to tackle the challenges of the knowledge society. This is a worthwhile direction for future research as there is a wide spectrum of knowledge that can be extracted from unstructured sources such as named entities, relations, or topics. For example, a Name Entity Recognition sensor would enable the identification of person entities such as staff members, events or other educational institutions. Also, improving field matching sensors is important since they guarantee that knowledge related to the same concept is correctly connected. Another issue to be addressed in future research is to better profile the computational requirements of our proposal. As an architecture, it is a high-level specification of components, their interfaces, and interactions [98]. As a consequence, overall computational requirements will depend on the implementation of the specific modules. Nonetheless, it is important to consider this issue since some approaches may introduce a prohibitive computation burden into the system.

Conflict of Interests

The authors report there are no conflicts of interest.

Acknowledgments

This research is supported by the University of Alicante, the Spanish Ministry of Science and Innovation, the Generalitat Valenciana, and the European Regional Development Fund (ERDF) through the following funding: At the national level, the following projects were granted: TRIVIAL (PID2021-122263OB-C22); and CORTEX (PID2021-123956OB-I00), funded by MCIN/AEI/10.13039/501100011033 and, as appropriate, by "ERDF A way of making Europe", by the "European Union" or by the "European Union NextGenerationEU/PRTR". At regional level, the Generalitat Valenciana (Conselleria d'Educacio, Investigacio, Cultura i Esport), granted funding for NL4DISMIS (CIPROM/2021/21). Moreover, it was backed by the work of two COST Actions: CA19134 - "Distributed Knowledge Graphs" and CA19142 - "Leading Platform for European Citizens, Industries, Academia, and Policymakers in Media Accessibility".

References

[1] S. John Walker, Big data: A revolution that will transform how we live, work, and think, 2014.

[2] X. Wu, X. Zhu, G.-Q. Wu, W. Ding, Data mining with big data, IEEE transactions on knowledge and data engineering 26 (2014) 97–107.

[3] H. Chen, R. H. Chiang, V. C. Storey, Business intelligence and analytics: from big data to big impact, MIS quarterly (2012) 1165–1188.

[4] D. V. Shah, J. N. Cappella, W. R. Neuman, Big data, digital media, and computational social science: Possibilities and perils, The ANNALS of the American Academy of Political and Social Science 659 (2015) 6–13.
[5] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Computer networks and ISDN systems 30 (1998) 107–117.
[6] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston, et al., The youtube video recommendation system, in: Proceedings of the fourth ACM conference on Recommender systems, ACM, pp. 293–296.
[7] D. Ferrucci, A. Levas, S. Bagchi, D. Gondek, E. T. Mueller, Watson: beyond jeopardy!, Artificial Intelligence 199 (2013) 93–105.
[8] B. Chandrasekaran, Generic tasks in knowledge-based reasoning: High-level building blocks for expert system design, IEEE expert 1 (1986) 23–30.
[9] M. Kevin, Machine learning: a probabilistic perspective, 2012.
[10] Q. V. Le, Building high-level features using large scale unsupervised learning, in: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, IEEE, pp. 8595–8598.
[11] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, A. T. Kalai, Man is to computer programmer as woman is to homemaker? debiasing word embeddings, in: Advances in Neural Information Processing Systems, pp. 4349–4357.
[12] P. Cimiano, A. Mädche, S. Staab, J. Völker, Ontology learning, in: Handbook on ontologies, Springer, 2009, pp. 245–267.
[13] K. Barker, B. Agashe, S. Y. Chaw, J. Fan, N. Friedland, M. Glass, J. Hobbs, E. Hovy, D. Israel, D. S. Kim, et al., Learning by reading: A prototype system, performance baseline and lessons learned, in: AAAI, volume 7, pp. 280–286.
[14] Z. Hu, P. Huang, Y. Deng, Y. Gao, E. Xing, Entity hierarchy embedding, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), volume 1, pp. 1292–1300.
[15] X. Li, T. Wang, Y. Pang, J. Han, J. Shi, Review of research on named entity recognition, in: Advances in Artificial Intelligence and Security, Springer International Publishing, 2022, pp. 256–267.
[16] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, J. Welling, Never-ending learning, Commun. ACM 61 (2018) 103–115.
[17] Z. Jin, T. Men, H. Yuan, Z. He, D. Sui, C. Wang, Z. Xue, Y. Chen, J. Zhao, CogKGE: A knowledge graph embedding toolkit and benchmark for representing multi-source and heterogeneous knowledge, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, 2022, pp. 166–173.
[18] G. Petasis, V. Karkaletsis, G. Paliouras, A. Krithara, E. Zavitsanos, Ontology population and enrichment: State of the art, in: Knowledge-driven multimedia information extraction and ontology evolution, Springer-Verlag, pp. 134–166.
[19] W. Litwin, L. Mark, N. Roussopoulos, Interoperability of multiple autonomous databases, ACM Computing Surveys (CSUR) 22 (1990) 267–293.
[20] A. P. Sheth, J. A. Larson, Federated database systems for managing distributed, heterogeneous, and autonomous databases, ACM Computing Surveys (CSUR) 22 (1990) 183–236.
[21] G. Fusco, L. Aversano, An approach for semantic integration of heterogeneous data sources, PeerJ Computer Science 6 (2020) e254.
[22] M. Lenzerini, Data integration: A theoretical perspective, in: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pp. 233–246.
[23] T. Adams, J. Dullea, P. Clark, S. Sripada, T. Barrett, Semantic integration of heterogeneous information sources using a knowledge-based system, in: Proc 5th Int Conf on CS and Informatics (CS&I'2000).
[24] C. Reynaud, J.-P. Sirot, D. Vodislav, Semantic integration of xml heterogeneous data sources, in: Proceedings 2001 International Database Engineering and Applications Symposium, pp. 199–208.
[25] C. Fellbaum, WordNet: An electronic lexical database, MIT press, 1998.
[26] S. Bergamaschi, S. Castano, M. Vincini, D. Beneventano, Semantic integration of heterogeneous information sources, Data & Knowledge Engineering 36 (2001) 215–249. Heterogeneous Information Resources Need Semantic Access.
[27] R. Vdovjak, G.-J. Houben, Rdf based architecture for semantic integration of heterogeneous information sources (2001).
[28] M. Buron, F. Goasdoué, I. Manolescu, M.-L. Mugnier, Obi-wan: ontology-based rdf integration of heterogeneous data, Proceedings of the VLDB Endowment 13 (2020) 2933–2936.
[29] A. Nentidis, K. Bougiatiotis, A. Krithara, G. Paliouras, Semantic integration of disease-specific knowledge, in:

IEEE 33rd International Symposium on Computer Based Medical Systems (CBMS)(2020, to appear).
[30] M. Rospocher, M. van Erp, P. Vossen, A. Fokkens, I. Aldabe, G. Rigau, A. Soroa, T. Ploeger, T. Bogaard, Building event-centric knowledge graphs from news, Web Semantics: Science, Services and Agents on the World Wide Web 37-38 (2016) 132–151.
[31] J. F. Sowa, et al., Knowledge representation: logical, philosophical, and computational foundations, volume 13, Brooks/Cole Pacific Grove, 2000.
[32] N. Guarino, Formal ontology, conceptual analysis and knowledge representation, International journal of human-computer studies 43 (1995) 625–640.
[33] O. Corcho, M. Fernández-López, A. Gómez-Pérez, Methodologies, tools and languages for building ontologies. where is their meeting point?, Data & Knowledge Engineering 46 (2003) 41–64.
[34] R. J. Brachman, H. J. Levesque, R. Reiter, Knowledge representation, MIT press, 1992.
[35] M. Chein, M.-L. Mugnier, Graph-based knowledge representation: computational foundations of conceptual graphs, Springer Science & Business Media, 2008.
[36] C. C. Aggarwal, H. Wang, et al., Managing and mining graph data, volume 40, Springer, 2010.
[37] D. Nadeau, S. Sekine, A survey of named entity recognition and classification, Lingvisticae Investigationes 30 (2007) 3–26.
[38] B. Liu, Sentiment analysis and opinion mining, Synthesis lectures on human language technologies 5 (2012) 1–167.
[39] E. Bingham, H. Mannila, Random projection in dimensionality reduction: applications to image and text data, in: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 245–250.
[40] T. Mikolov, W.-t. Yih, G. Zweig, Linguistic regularities in continuous space word representations, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 746–751.
[41] M. Lubani, S. A. M. Noah, R. Mahmud, Ontology population: Approaches and design aspects, Journal of Information Science 45 (2019) 502–515.
[42] U. Hahn, M. Romacker, The syndikate text knowledge base generator, in: Proceedings of the first international conference on Human language technology research, Association for Computational Linguistics, pp. 1–6.
[43] H. Alani, S. Kim, D. E. Millard, M. J. Weal, W. Hall, P. H. Lewis, N. R. Shadbolt, Automatic ontology-based knowledge extraction from web documents, IEEE Intelligent Systems 18 (2003) 14–21.
[44] P. Buitelaar, P. Cimiano, S. Racioppa, M. Siegel, Ontology-based information extraction with soba, in: Proceedings of the International Conference on Language Resources and Evaluation (LREC).
[45] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, S. Slattery, Learning to construct knowledge bases from the world wide web, Artificial intelligence 118 (2000) 69–113.
[46] H. Stoermer, I. Palmisano, D. Redavid, L. Iannone, P. Bouquet, G. Semeraro, Contextualization of a RDF Knowledge Base in the VIKEF Project, Springer Berlin Heidelberg, pp. 101–110.
[47] C. Brewster, F. Ciravegna, Y. Wilks, User-centred ontology learning for knowledge management, Natural Language Processing and Information Systems (2002) 203–207.
[48] S.-S. Kim, J.-W. Son, S.-B. Park, S.-Y. Park, C. Lee, J.-H. Wang, M.-G. Jang, H.-G. Park, Optima: An ontology population system, in: 3rd Workshop on Ontology Learning and Population (July 2008).
[49] N. Weber, P. Buitelaar, Web-based ontology learning with isolde, in: Proc. of the Workshop on Web Content Mining with Human Language at the International Semantic Web Conference, Athens GA, USA, volume 11.
[50] P. Buitelaar, M. Sintek, Ontolt version 1.0: Middleware for ontology extraction from text, in: Proc. of the Demo Session at the International Semantic Web Conference.
[51] F. M. Suchanek, G. Ifrim, G. Weikum, Leila: Learning to extract information by linguistic analysis, in: Proceedings of the 2nd Workshop on Ontology Learning and Population: Bridging the Gap between Text and Knowledge, pp. 18–25.
[52] P. Cimiano, J. Völker, text2onto, in: International Conference on Application of Natural Language to Information Systems, Springer, pp. 227–238.
[53] O. Etzioni, M. Cafarella, D. Downey, S. Kok, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, A. Yates, Web-scale information extraction in knowitall:(preliminary results), in: Proceedings of the 13th international conference on World Wide Web, ACM, pp. 100–110.
[54] E. Drymonas, K. Zervanou, E. G. Petrakis, Unsupervised ontology acquisition from plain texts: the OntoGain system, in: International Conference on Application of Natural Language to Information Systems, Springer, pp. 277–287.
[55] D. Faure, T. Poibeau, First experiments of using semantic knowledge learned by asium for information extraction task using intex, in: Proceedings of the ECAI workshop on Ontology Learning.
[56] S. Castano, S. Espinosa, A. Ferrara, V. Karkaletsis, A. Kaya, S. Melzer, R. Möller, S. Montanelli, G. Petasis, Ontology dynamics with multimedia information: The boemie evolution methodology, in: International Workshop

on Ontology Dynamics (IWOD-07), p. 41.
[57] S. Estevez-Velarde, Y. Gutierrez, A. Montoyo, A. Piad-Morffis, R. Munoz, Y. Almeida-Cruz, Gathering object interactions as semantic knowledge, in: Proceedings on the international conference on artificial intelligence (icai), The Steering Committee of The World Congress in Computer Science, Computer . . . , pp. 363–369.
[58] S. Estévez-Velarde, Y. Gutiérrez, Y. Almeida-Cruz, A. Montoyo, General-purpose hierarchical optimisation of machine learning pipelines with grammatical evolution, Information Sciences (2020).
[59] J. Fernández, Y. Gutiérrez, J. M. Gómez, P. Martínez-Barco, Social rankings: análisis visual de sentimientos en redes sociales, Procesamiento del Lenguaje Natural 55 (2015) 199–202.
[60] A. Balahur, J. M. Hermida, A. Montoyo, R. Muñoz, EmotiNet: A Knowledge Base for Emotion Detection in Text Built on the Appraisal Theories, Springer Berlin Heidelberg, pp. 27–39.
[61] A. Montoyo, P. MartíNez-Barco, A. Balahur, Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments, 2012.
[62] J. M. Hermida, S. Meliá, J.-J. Martínez, A. Montoyo, J. Gómez, Developing semantic rich internet applications with the s m 4ria extension for oide, in: International Conference on Web Engineering, Springer, pp. 20–25.
[63] S. Staab, R. Studer, Handbook on ontologies, Springer Science & Business Media, 2010.
[64] N. Choi, I.-Y. Song, H. Han, A survey on ontology mapping, ACM Sigmod Record 35 (2006) 34–41.
[65] A. B. Y. An, J. Mylopoulos, Building semantic mappings from databases to ontologies, volume 21st National Conference on Artificial Intelligence (AAAI 06).
[66] N. F. Noy, M. A. Musen, The prompt suite: interactive tools for ontology merging and mapping, International Journal of Human-Computer Studies 59 (2003) 983–1024.
[67] N. KONSTANTINOU, D.-E. SPANOS, N. MITROU, Ontology and database mapping: A survey of current implementations and future directions.
[68] C. P. de Laborda, S. Conrad, Bringing relational data into the semantic web using sparql and relational.owl.
[69] P. Vassiliadis, A survey of extract–transform–load technology, International Journal of Data Warehousing and Mining (IJDWM) 5 (2009) 1–27.
[70] A. Orlowski, Wikipedia founder admits to serious quality problems, The Register 18 (2005).
[71] M. Hu, E.-P. Lim, A. Sun, H. W. Lauw, B.-Q. Vuong, Measuring article quality in wikipedia: models and evaluation, in: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, ACM, pp. 243–252.
[72] P. Exner, P. Nugues, Using semantic role labeling to extract events from wikipedia, in: Proceedings of the workshop on detection, representation, and exploitation of events in the semantic web (DeRiVE 2011). Workshop in conjunction with the 10th international semantic web conference, pp. 23–24.
[73] A. Gattani, D. S. Lamba, N. Garera, M. Tiwari, X. Chai, S. Das, S. Subramaniam, A. Rajaraman, V. Harinarayan, A. Doan, Entity extraction, linking, classification, and tagging for social media: a wikipedia-based approach, Proceedings of the VLDB Endowment 6 (2013) 1126–1137.
[74] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, O. Etzioni, Open information extraction from the web., in: IJCAI, volume 7, pp. 2670–2676.
[75] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.
[76] F. Giunchiglia, M. Fumagalli, Teleologies: Objects, actions and functions, in: International Conference on Conceptual Modeling, Springer, pp. 520–534.
[77] L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, N. Schneider, Abstract meaning representation for sembanking, in: Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pp. 178–186.
[78] T. Hofmann, Probabilistic latent semantic indexing, in: ACM SIGIR Forum, volume 51, ACM, pp. 211–218.
[79] Q. Guo, W. Wu, D. Massart, C. Boucon, S. De Jong, Feature selection in principal component analysis of analytical data, Chemometrics and Intelligent Laboratory Systems 61 (2002) 123–132.
[80] J. Turian, L. Ratinov, Y. Bengio, Word representations: a simple and general method for semi-supervised learning, in: Proceedings of the 48th annual meeting of the association for computational linguistics, Association for Computational Linguistics, pp. 384–394.
[81] M. Fernández-López, A. Gómez-Pérez, N. Juristo, Methontology: from ontological art towards ontological engineering (1997).
[82] S. Haykin, N. Network, A comprehensive foundation, Neural networks 2 (2004) 41.
[83] N. F. Noy, M. A. Musen, et al., Algorithm and tool for automated ontology merging and alignment, in: Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-00). Available as SMI technical report SMI-2000-0831.
[84] N. Aussenac-Gilles, M.-P. Jacques, Designing and evaluating patterns for ontology enrichment from texts, in: International Conference on Knowledge Engineering and Knowledge Management, Springer, pp. 158–165.
[85] E. Blomqvist, Ontocase-automatic ontology enrichment based on ontology design patterns, in: International

Semantic Web Conference, Springer, pp. 65–80.
[86] P. Shvaiko, J. Euzenat, Ontology matching: state of the art and future challenges, IEEE Transactions on knowledge and data engineering 25 (2013) 158–176.
[87] P. Ciccarese, M. Ocana, L. J. Garcia Castro, S. Das, T. Clark, An open annotation ontology for science on web 3.0, Journal of Biomedical Semantics 2 (2011) S4.
[88] N. Chalortham, P. Leesawat, M. Buranarach, T. Supnithi, Ontology development for pharmaceutical tablet production expert system, in: Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008. ECTI-CON 2008. 5th International Conference on, volume 1, IEEE, pp. 205–208.
[89] C. Alberti, D. Andor, I. Bogatyy, M. Collins, D. Gillick, L. Kong, T. Koo, J. Ma, M. Omernick, S. Petrov, C. Thanapirom, Z. Tung, D. Weiss, Syntaxnet models for the conll 2017 shared task, CoRR abs/1703.04929 (2017).
[90] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, International Journal of Computer Vision 115 (2015) 211–252.
[91] J. Brank, D. Mladenic, M. Grobelnik, Gold standard based ontology evaluation using instance assignment, in: Workshop on Evaluation of Ontologies for the Web, EON, Edinburgh, UK.
[92] A. Lozano-Tello, A. Gómez-Pérez, Ontometric: A method to choose the appropriate ontology, Journal of database management 2 (2004) 1–18.
[93] F. Corcoglioniti, M. Rospocher, A. P. Aprosio, Frame-based ontology population with pikes, IEEE Transactions on Knowledge and Data Engineering 28 (2016) 3261–3275.
[94] I. Gurevych, R. Malaka, R. Porzel, H.-P. Zorn, Semantic coherence scoring using an ontology, in: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, Association for Computational Linguistics, pp. 9–16.
[95] J. Brank, M. Grobelnik, D. Mladenić, A survey of ontology evaluation techniques (2005).
[96] A. O. Al-Sulttani, M. Al-Mukhtar, A. B. Roomi, A. A. Farooque, K. M. Khedher, Z. M. Yaseen, Proposition of new ensemble data-intelligence models for surface water quality prediction, IEEE Access 9 (2021) 108527–108541.
[97] H. A. Afan, A. Ibrahem Ahmed Osman, Y. Essam, A. N. Ahmed, Y. F. Huang, O. Kisi, M. Sherif, A. Sefelnasr, K.-w. Chau, A. El-Shafie, Modeling the fluctuations of groundwater level by employing ensemble deep learning techniques, Engineering Applications of Computational Fluid Mechanics 15 (2021) 1420–1439.
[98] R. S. Pressman, Software engineering: a practitioner's approach, Palgrave macmillan, 2005.

Dear Editor and Reviewers,

First, thank you very much for your comments, which we consider to have greatly improved the quality of our work. This letter contains each reviewer's comments as well as our explanation of how they were addressed. Moreover, the full manuscript was reviewed to improve writing.

Sincerely,

Authors

Reviewer #6:

I think the authors have addressed properly the comments given by the reviewers. The paper is ready to be accepted.

Reviewer #7:

The manuscript has been substantially improved.

Reviewer #8:

The authors have made significant changes to the paper, addressing a significant portion of the reviewers' comments. However, there are still severe limitations in some especially important aspects, which are listed below:

1) **The abstract and the introduction can be clearly improved.**

R/ Both sections were reworked. In the abstract, we clearly explain:

- problem: the semantic integration from heterogeneous sources.
- main contribution: a plugin-based architecture addressing the challenges related to knowledge extraction, data integrity, and scalability that are inherent in the semantic integration of heterogeneous data.
- results: architecture and case study demonstrating the implementation of the architecture.

The Introduction was reviewed. We clarified some of the main motivations of our proposal, as well as the core challenges faced by a semantic integration process.

2) **The new introduction fails to motivate the utility of semantic integration of heterogeneous data and to clearly define the contributions of the work compared to previous approaches.**

R/ We reviewed the Introduction to better motivate our proposal as well as to highlight our contributions. Please, see answers to comment 1.

3) **The contributions are listed, but the necessary context to understand their originality is missing.**

R/ We revised the Introduction to make clear the main challenges faced by semantic integration of heterogeneous data and how our proposal has addressed them. Also, we reviewed the State of the Art to better explain the context and antecedents of our research.

4) **The new introduction includes two subsections on motivation and objectives that would be more convenient to merge and restructure appropriately, organizing the ideas clearly for the rest of the article to be easily readable.**

R/ The section has been rewritten according to the suggestions.

5) **The state of the art review is diffuse, generic, and appears to be more focused on referencing works than on contextualizing them and clarifying the relationship and differences with the approach proposed in this paper. After reading the article, I share the impression of a previous referee who mentioned that the literature was insufficient, requesting an analysis of more recent related works. Although the authors claim to have updated their references, the new Related Work section includes some new references but lacks the requested analysis. A paragraph has been added at the end of the section, which seems insufficient given the identified deficiency in the initial review. The authors should incorporate related approaches regarding methodologies and applications of semantic integration of heterogeneous data, analyzing in detail the identified problems and how their research addresses them to a greater or lesser extent. Right now, the literature review is insufficient.**

R/ We appreciated the opportunity to address this issue. We reworked the State of the Art section. Now, we differentiated between (i) works dealing with the same goal as us, the semantic integration of heterogeneous data, (ii) works related to knowledge representation and reasoning, (iii) the use of machine learning to extract knowledge, and (iv) the process of knowledge discovery.

For (i) a new subsection "Heterogenous Data and Knowledge Integration" was added. There, we analyzed the approaches, discussing potential drawbacks, limitations, and how our proposal may overcome these issues.

Sections related to points (ii), (iii), and (iv) were kept providing context of developments that are not fully dedicated to the semantic integration (as a holistic process) but that address sub-problems within this area.

6) **Figure 3 is unnecessary. The information represented is already covered in the text, so the graphical representation does not provide any relevant insights.**

R/ …it's deleted

Reviewer #9:

Following are review comments:

1) **Abstract to have key conclusion at the last portion of the abstract**

R/ The Abstract was rewritten to better explain (i) the problem we addressed that is the semantic integration of heterogeneous sources; (ii) our main contribution, a plugin-based architecture tackling challenges faced by the task of semantic integration of heterogenous sources: knowledge extraction, data integrity, and scalability; (iii) results and key conclusions.

2) **Keywords to be organized in alphabetical order. Author may add upto two key words considering size of manuscript influencing total manuscript.**

R/ …Updated!

3) **Following references to be cited**

R/ Thanks for the suggestions. We reviewed the State of the Art section to better contextualize our proposal. We include a new subsection "2.1 Heterogenous Data and Knowledge Integration" with a critical review of the provided references as well as others that are very relevant to our proposal.

4) **Proposal section can be rewritten as Methodology with flow diagram of complete research.**

R/ Figure 1 represents the main workfow of the proposal. Section label updated!

5) **Objective pointwise to be discussed in discussion and conclusion session so that readers understand that each aspect of objective is achieved/ not achieved or further research is required.**

R/ Section 6 Discussion and Conclusions have been reorganized and improved to make explicit the main challenges addressed and how our proposal solved them.