INSTITUTO TECNOLÓGICO
Y DE ESTUDIOS SUPERIORES DE OCCIDENTE

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación el 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática

Doctorado en Ciencias de la Ingeniería

MEJORA DEL APRENDIZAJE ONTOLÓGICO A TRAVÉS DE TÉCNICAS DE SEMÁNTICA INTERPRETATIVA

Tesis que para obtener el grado de
Doctor en Ciencias de la Ingeniería
presenta: Luis Miguel Escobar Vega

Director de tesis: Dr. Víctor Hugo Zaldívar Carrillo
Co-director de tesis: Dr. Iván Vilallón Turrubiates

Tlaquepaque, Jalisco. Junio de 2021

TITULO: **Mejora del aprendizaje ontológico a través de técnicas de semántica interpretativa**

AUTOR: Luis Miguel Escobar Vega
Ingeniero en Computación (Universidad del Valle de Atemajac, México)
Maestro en Computación (Universidad del Valle de Atemajac, México)

DIRECTOR DE TESIS: Víctor Hugo Zaldívar Carrillo
Departamento de Electrónica, Sistemas e Informática, ITESO
Ingenierio en Sistemas Computacionales (ITESO, México)
Maestría en Informática con Especialidad en Inteligencia Artificial (Universidad de Montpellier II, Francia)
Doctor en Ciencias con especialidad en Inteligencia Artificial y Sistemas Cognitivos (Universidad de Montpellier II, Francia)

NÚMERO DE PÁGINAS: XXVII, 154

ITESO – The Jesuit University of Guadalajara

Department of Electronics, Systems and Informatics
Doctoral Program in Engineering Sciences

USING INTERPRETIVE SEMANTICS TECHNIQUES TO ENHANCE ONTOLOGY LEARNING

Thesis to obtain the degree of
Doctor in Engineering Sciences
Presents: Luis Miguel Escobar Vega

Thesis Director: Dr. Víctor Hugo Zaldívar Carrillo
Thesis Co-director: Dr. Iván Vilallón Turrubiates

Tlaquepaque, Jalisco, Mexico

June 2021

TITLE: **Using interpretive semantics techniques to enhance ontology learning**

AUTHOR: Luis Miguel Escobar Vega
Bachelor's degree in computer engineering (University of Valle de Atemajac, Mexico)
Master's degree in computer engineering (University of Valle de Atemajac, Mexico)

THESIS DIRECTOR: Víctor Hugo Zaldívar Carrillo
Departament of Electronics, Systems, and Informatics, ITESO
Bachelor's degree in Computer Systems Engineering (ITESO, México)
Master's degree in computer science with specialty in artificial intelligence (Montpellier II University, Francia)
Ph.D degree sciences specialized in artificial intelligence and cognitive systems (Montpellier II University, Francia)

NUMBER OF PAGES: XXVII, 154

To my parents Miguel and Ma.

Resumen

A medida que el desarrollo de asistentes inteligentes se integra a mercados de uso masivo, las técnicas tradicionales de validación de los sistemas de pregunta respuesta, usadas para resolver preguntas, resultan ineficaces para lograr una cobertura funcional completa del sistema. Las complejidades inherentes del lenguaje natural, y del diálogo conversacional, crean grandes desafíos para garantizar que los asistentes virtuales funcionen adecuadamente en todas las condiciones del proceso del habla y la comprensión. Adicionalmente, cada vez hay un número mayor de modelos de lenguaje natural entrenado en los asistentes virtuales. Una parte importante de ellos corresponde a procesos estadísticos. Mejoras en los datasets, en las técnicas del procesamiento del lenguaje natural y en la velocidad de procesamiento, han permitido mejores resultados donde el *Score* rebasa el 90% de comprensión. Los efectos de la falta de interpretación del texto pueden crear múltiples problemas de comprensión al resolver una pregunta. Estos problemas se agravan cuando el dataset se encuentra con contextos nuevos o diferentes en los cuales no ha sido entrenado. Resulta evidente que los retos de la comprensión del significado están en constante aumento. Uno de los mayores desafíos radica en el proceso de recuperación de información donde se utilizan técnicas para extraer elementos clave del lenguaje. Ajustar la recuperación de información para extraer elementos clave del lenguaje es una tarea demandante en la validación de elementos semánticos. Las propuestas actuales para identificación de elementos clave del significado requieren información masiva, ya que se basan en métodos estadísticos entrenados exhaustivamente, convirtiendo el proceso de recuperación de información en una tarea prolongada y prácticamente imposible frente a contextos nuevos. En esta tesis doctoral se propone combinar métodos de semántica interpretativa, de aprendizaje de ontologías y de similitud semántica con funciones estadísticas apropiadas, de manera que la extracción de elementos semánticos de un texto se lleve a cabo con mayor eficiencia. La evaluación de los métodos es realizada mediante mediciones de laboratorio en plataformas realistas como asistentes virtuales. Los resultados obtenidos demuestran la eficacia de los métodos propuestos, así como una mejora sustancial en desempeño, con respecto a la práctica actual.

Summary

As intelligent virtual assistant scales to the mass market, traditional validation techniques for question-answering systems become inappropriate to get full functional coverage of the system. Natural language and conversational dialog inherent complexities introduce design challenges to guarantee process, talk, and understanding performance. Besides, there is and an increasing number of training language models in question-answering systems. A significant portion of them corresponds to the statistic-based language model. Improvements in datasets, natural language processing techniques, and processing speed have allowed better data rates to scale beyond 90% of the *Score*. Some effects of the lack of interpretation can create multiple understanding integrity problems in solving a question. This problem is aggravated when the model faces a new and different context from that used in the training process. Challenges for meaning comprehension are continuously increasing. Therefore, information retrieval processes extract key elements of the language that can be critical for making more useful question-answering systems. Using appropriate information retrieval techniques to extract critical elements that can be used to create new knowledge structures is a significant challenge. The combination of information retrieval and ontology learning can be a very consuming validation task. Typical practices in question-answering systems construction are statistic-based. Consequently, they require massive datasets to train their models, making the information retrieval process too lengthy and prohibitive when the model faces new contexts. In this doctoral dissertation, the combination of interpretive semantics, semantic similarity, and ontology learning methods with suitable statistical functions is proposed to improve the efficiency of extracting semantic elements from a text. The proposed methods are implemented in a software tool, and its performance is evaluated on real question-answering platforms such as virtual assistants. The results show both the efficiency of the proposed methods and significant improvements when compared to state-of-the-art practices.

Acknowledgements

The author wishes to express his sincere appreciation to Dr. Víctor Hugo Zaldívar Carrillo, professor of the Department of Electronics, Systems, and Informatics at ITESO, for his encouragement, expert guidance, and keen supervision as doctoral thesis director throughout the course of this work. The author offers his gratitude to Dr. Iván Villalón Turrubiates, from ITESO, for his support as doctoral thesis co-director during the development of this work. He also thanks Dr. Rogelio Dávila Pérez, Dr. José Francisco Cervantes Álvarez, Dra. Mildreth Isadora Alcaraz Mejía, members of his Ph.D. Thesis Committee, for their interest, assessment, and suggestions. The author gratefully acknowledges the financial assistance through the scholarship number 399053 granted by the Consejo Nacional de Ciencia y Tecnología (CONACYT), Mexican Government. Finally, special thanks are due to my family: my beautiful beloved Arelí and my children Santiago and Alaia for their understanding, patience, and continuous loving support.

Contenido

Contents

List of Figures

List of Tables

List of acronyms

Introduction

Nowadays, the volume of information a person requires to make decisions is higher than in the past. People ask questions about personal, professional, day-to-day matters or to confirm something they do not remember. The Internet has expanded into an almost unlimited information base. Every time a question is asked, there is more than one source of information to rely on, and there is limited time to get answers. Furthermore, what is learned must be confirmed and validated. Moreover, it must be up to date, considering that information changes and evolves continuously.

Since its foundation, the Artificial intelligence community has searched to develop tools to facilitate ordinary's people access to knowledge. For instance, in the middle of the last century, the intelligent personal assistant (IPA) was created [Simmons-67]. Since then, IPAs use has been growing exponentially. Mainly due to the emergence of personal smart devices. According to the World Economic Forum[1], by 2030, more than one trillion devices will be used worldwide

The development of tools to manage extensive information enabled the applications to use higher knowledge. It contributed to make the devices smarter [Russell-10]. On the other hand, the improvement of internet broadband and cloud services lowering costs aim to make the Internet affordable for everyone. IPAs can interact with people in various forms through commands, alerts, recommendations, queries, or information filters. Nevertheless, IPAs that operate with questions and answers are more significant for human-computer interaction (HCI) because they can establish dialogues with people through questions and answers [Dourish-14]. Question answering system (QAS) also represents one of the most significant challenges of natural language comprehension.

The primary function of QAS is to analyze the questions made by one person in a natural language, parse them into some data structure, and then search among different data sources to obtain the best option to answer the question. In addition, common QASs determine some issues using clues. Afterward, QAS searches for words that are classified and pondered using statistical techniques. A QAS works with high scores when there is plenty of information available to learn from it. Several Text-based QAS extract their features with statistical methods. Statistics will

[1]Digital Transformation, World Economic Forum. Jul. 8, 2020, http://reports.weforum.org/digital-transformation/surviving-digital-disruption.

undoubtedly remain an essential element for finding information. Nevertheless, they need to be helped by semantic techniques to find meaning in their context. This work has turned to formal semantics to enhance the purely quantitative approach.

The QASs have adopted several techniques to find answers. Recent proposals of QAS based on knowledge base (KB) have had successful results [Ferrucci-12], [Xu-14], [Baudis-15], [Kuznetsov-16], [Hakimov-17] and [Diefenbach-18], the release of ontological standards such as web ontology language (OWL)[2] provided by the World Wide Web Committee has accelerated their use. OWL offers benefits to QAS since it provides a symbolic knowledge representation structure and mechanisms for logical inferences through ontologies. Although knowledge representation increases the QAS score significantly, it is only plausible to use structured information.

An essential application of ontologies is to provide semantics and domain knowledge for data. Notwithstanding that, it is questionable if the ontologies are the best way to maintain a linguistic structure, given that defining concepts and their relations is far from representing the language. Thus, its use is limited to axioms. Furthermore, ontologies can be represented in several structures such as semantic networks, triplets, n-tuples, semantic graphs, and semantic boxes, but none can fully represent the semantic expressiveness of ontologies.

Ontology learning (OL) supports the construction of ontologies and populating them with instantiations of both concepts and relations. Originally OL was described as the acquisition of a domain model from data. Cimiano's work [Cimiano-06] on OL has focused on creating and populating ontologies using natural language processing (NLP), formal concept analysis (FCA) [Wille-82], and machine learning (ML) techniques to extract features.

The problem of OL is that ontology always reflects a way of conceptualizing the world. In contrast, the results of the OL algorithm that learn from a dataset reflect the peculiarities of the dataset in question [Lehmann-15]. We propose a new approach to solve this problem using interpretive semantics (IS) methods, using isotopies (repetition of the same meaning), including contextual meaning, and reflecting more real-world concepts. Interpretive semantics (IS) has the potential to increase the performance of QAS approaches that uses OL.

Information retrieval (IR) and QAS also face the vocabulary mismatch problem. In fact, due to language variation, in most cases, people will ask for information stated on a web page in several

[2]World Wide Web Consortium, W3C. Sep. 10, 2018, https://www.w3.org.

different ways making this correspondence challenging to be directly observed. Identifying and correctly interpreting paraphrases about some knowledge is an essential issue within question answering. For this reason, the identification and correct interpretation in the paraphrase of some texts is a challenge, essential to discover answers to the questions.

Description logic (DL) is a language for exploiting ontologies. It is composed of collections of knowledge representation languages historically related to semantics. DL is based on ontology models, and logical formalism is restricted to decidable fragments of first-order logic (FOL). DL is equipped with formal semantics, allowing humans and computer systems to exchange DL ontologies, making it possible to use logical deduction to infer additional knowledge from the facts stated explicitly in an ontology.

In this dissertation, we examine how to work with DL to obtain high results in the Score of the QASs. We use semantic queries to create inferences in fragments of text related to semantic evidence for a probable answer[3]. Statistical methods will remain a critical tool for finding information. Statistics methods need to be augmented by formal semantic models. The closer interaction between statistical methods and semantic models, the more suitable comprehension of natural language, and the closer we are of models resembling human understanding.

There are a plethora of models for the treatment of text semantics. We explore Tesnière's theory [Tesniere-15] as a starting point in analysis dependency syntax and grammar. Universal grammar (UG) by [Montague-70] explores a logic-based approach for the treatment of language. Combinatory categorial grammar (CCG) [Steedman-01] and Blackburn [Blackburn-05], among others, provide primitive rules for constructing basic structures to represent knowledge. IS identify semantic units and describe operations that govern meaning in a text by a given interpreter. In IS, text interpretation is flexible and, at the same time, gives a central place to the subjectivity in the condition of the linguistic object. In this work, we use interpretive semantics methods to reinforce the retrieval process in the QASs.

Text classification (TC) is an active area in the data processing. Lately, valuable and essential tools have been developed to improve the search for semantic patterns in information, favoring the interpretation and obtaining better precision in identifying the meaning. In this dissertation, classification is used to detect semantic evidence that helps measure the similarity and differences

[3] e_1 and e_2 are objects and y a relation, the triplet states that there is a relation y from object e_1 to object e_2.

between words using semantic traits.

To advance the development of QAS, we start by developing complex alignment benchmarks for real-world ontologies. Then, we build an OL model to extract semantic features from answers, and then we create a QAS that uses IS to enhance the OL process.

One of the principal objectives of this thesis is to improve the structure and the content of the ontologies. The chapters of this work treat real troublesome in transforming the information to knowledge, the text classification, and the question and answer systems. The proposals gradually solve the problems with interpretive semantics methods as a novel solution to contextual interpretation and understanding, common to current statistical methods. During the development of this work, it was found that the importance of semantic analysis to obtain good results is increasingly manifested. Furthermore, many of the contributions of this work were achieved in information retrieval, where current statistical methods were combined with interpretive semantics.

The main contributions are: (a) Semantic structures were created that operate as characteristics in the text classification processes and avoid deficiencies in the request for contextual information in traditional statistical approaches, (b) Methods were developed that provide robust and compact ontologies, facilitating the incorporation of logical reasoners and improving the coverage of inferences. (c) A formal theory was implemented to interpret the meaning in the text. This proposal made it possible to reach a primary semic interpretation and enrich the relationships of the text's elements. (d) A QAS approach was proposed that combines semantic techniques in the response extraction process, which improved the responses' scores. Finally (e), our main contribution was combining ontological learning with interpretive semantics to generate higher quality ontologies, that is, more compact and complete ontologies.

This doctoral dissertation is organized as follows: Chapter 1 reviews the historical framework and basic concepts on ontologies, semantic structures, and a summary of question answering systems. Chapter 2 proposes methods to transform invoices into semantic models, and modeling of ontological invoices is built. In Chapter 3, a study of text classification techniques and semantic classifiers is presented. The statistical phenomenon is explained, its impact on classifiers, and the concept of semantic evidence is outlined. Chapter 4 proposes formalization and a computing model for interpretive semantics. Experimentation is carried out to improve information retrieval with interpretive semantics methods. Chapter 5, combines ontology learning methods with interpretive

analysis and then uses this new approach to text classification and question answering system to improve their scores. In Chapter 6, a hybrid ontology learning base question answering system model is presented. Interpretive semantics methods enrich the data in the retrieval processes to find answers with high scores. In General conclusions, the most relevant contributions and comments on this doctoral thesis are summarized, discussing the global results of the proposed interpretive semantics techniques and the new objective functions to optimize classification and question answering system. Finally, some opportunities for future research are briefly described.

1. Related work

In this chapter, we examine the body of literature related to our dissertation. The broader aim is to increase the quality of ontologies created with ontology learning. We cover related work in knowledge structures and knowledge representation from philosophy to linguistics to formal semantics and logic. We pay special attention to interpretive semantics and description logic and conclude with a review of work about question answering system.

1.1. Philosophy and knowledge

Knowledge is a human sub-product result of our settlement in the universe. It is not something that has been designed to be a final product, but it has been the means of solving a particular problem. In other words, knowledge is usually the result of life experiences that have been maintained over time. There is no consensus on the nature of knowledge [Jones-52]. Great thinkers of the "classical Greek" have sought to define knowledge. Plato observed that people get ideas to explain further and to argue the reasons for different subjects. He considered that ideas existed beside things. They are not just concepts or mind depictions, as the current meaning of idea suggests it. These realities exist independently of things. Furthermore, they are the reality itself in contrast with things that are less real. Every idea is unique, eternal, immutable, and inalterable, just accomplished by intelligence. It is worth to say that it is a non-sensible and intelligible reality [Plato-01]. Aristotle[1] dealt with this subject in his metaphysics and defined ontology as the science of "being qua being."

1.1.1 Knowledge as an emergent property of evolutionary systems

In the late twentieth century, Karl Popper introduced the notion of three ontological domains or worlds of objective knowledge [Popper-72], which has much in common with Plato's theory of forms or ideas. Afterward, he adopted the concept where "knowledge is solutions to problems" or at least claims towards solutions. He argued that knowledge emerges in living things as they adapt to the world. In his most complete explanation referred to this as his "general theory of

[1] "Qua" means "by virtue of what it is." "What it is" is the essence under a context.

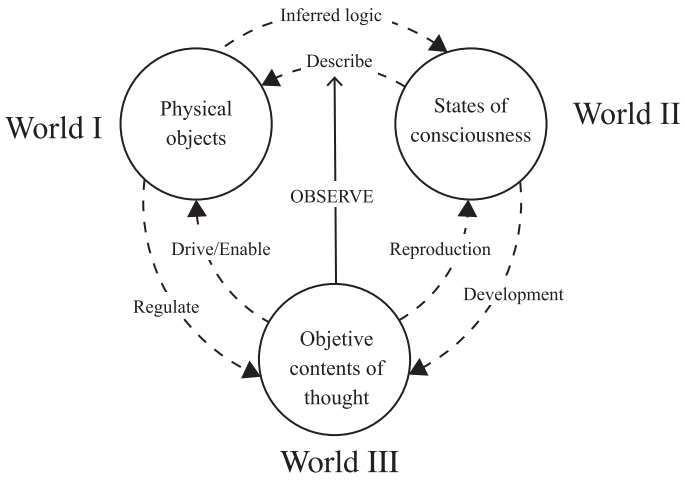


Fig. 1.1 Popper: Knowledge in the three worlds ontology.

evolution" [Popper-72]. The ontology of Popper consists of three domains Fig. 1.1, or as he called them "Worlds," that interact with each other: World I, the world of physical objects or physical states; World II, the world of states of consciousness, or states of mind, or perhaps of behavioral dispositions to act; and World III, the world of objective contents of thought, especially of scientific and poetic ideas and of works of art. The classification of knowledge in these worlds provides a tool to categorize it, identifies how knowledge interacts with other knowledge, understands its evolution, and improves testing observations in objective knowledge. Knowledge emerges when there are three ontological domains involved. Popper argued that knowledge claims might be aggregated and transformed. A theory referring to World I can be constructed by people in World II and be expressed and shared in the form of World III content [Hall-05], through iterated cycles of hypothesizing solutions, and testing and criticizing them to eliminate errors. Knowledge claims asserted in World II or World III can move toward correspondence with the reality of World I, as Popper [Popper-72] explained in his "general theory of evolution." These interactions make the difference between these three worlds; they are as crucial as the epistemic distinctions between them. It is these interactions that differentiate Popper's approach from Plato's static method and lays at the heart of the human understanding of abstract objects that are textual representations of knowledge in World III. The evolution of knowledge emerges in living things as they adapt to the world. In his most complete explanation, Popper referred to this as his "general theory of evolution." These interconnected ideas formed the basis of Popper's "general theory of evolution" and the "growth of knowledge" that takes

place in living entities. Popper's "general theory of evolution" is expressed as his "tetradic schema,"

$$P_1 \rightarrow TT \rightarrow EE \rightarrow P_2 \tag{1–1}$$

where P is a problem, TT is a tentative theory, and EE is error-elimination. Popper notes that there may be multiple tentative solutions to a problem, and all are filtered through the EE process. To emphasize the tetradic schema's recursiveness, [Hall-05] changed Popper's P_1 to P_n and P_2 to $P_n + 1$. Hall also drew in a return loop to remind readers the schema itself is iterated endlessly in a not quite closed cycle. Having completed a cycle (or generation of selection), the problem's situation is changed by prior solutions, and the new tentative solutions may not be the same. In conclusion, structures that stored knowledge are only traces that show their presence in a timeline. It is evident and unavoidable the growth of the knowledge, and that the expansion does not also develop just in an only world of Popper, but that nourishes under the influence of other worlds, e.g., the context. Therefore, this suggests that a large part of the knowledge is lost in the iterations and that if we want to recover it, we must look in the other worlds for fragments of knowledge that may no longer exist. Interpretation is a central issue in this work, particularly semantics. Interpretation deals with the search of these lost fragments through the assumption of textual meanings. Nevertheless, these assumptions will always be biased by a human or a computer interpreter.

1.1.2 Knowledge representation and reasoning

The human intellect is the most complex phenomenon studied in computer science, particularly in artificial intelligence. Its complexity involved is due to its conditioning by knowledge. To work with intelligence, it is necessary to represent and build a knowledge structure so that a machine can understand and process it to solve a problem. Brachman and Levesque [Brachman-04] proposed three key concepts: knowledge, representation, and reasoning. Knowledge is the nature of the propositions, and representation is a relationship between two domains, where the first is meant to "stand for" or take the place of the second. Finally, the reasoning is a form of calculation, not unlike arithmetic, but over symbols standing for propositions rather than numbers. According to Baader [Baader-10], one of the objectives of the DL is to create ontologies, link them through a standardized language, such as FOL, and add reasoning procedures to give a semantic environment. However,

this is possible only if Popper's knowledge structure is considered, and to apply the reasoning is necessary to use objective propositions. Without logic, knowledge representation is vague, with no criteria for determining whether statements are redundant or contradictory. Without an ontology, the terms and symbols are ill-defined and confusing. Knowledge representation (KR) is the application of logic and ontology [Cope-11]. For the Semantic Web to succeed, the expressive power of the logic added to its markup languages must be balanced against the resulting computational complexity. As has been described in the last paragraphs, the semantic web has concrete bases; many researchers have contributed to the history, but not all the knowledge has been carried into commercial solutions. This became a problem, and it is the central reason for this research, the lack of semantic tools in commercial applications. Commercial applications currently look more like silos of information, different from semantic web applications where information is distributed over the immensity of the Internet. Besides this, the structure of an application is not appropriate for working between semantic and information structures. Finally, the nature of the information does not allow interpreters to be expressed in formal language easily, making it challenging to apply reasoning methods. A better approach needs to involve many areas, not only the Semantic Web. Topics like knowledge management, logic description, natural language processing, and big data are essential key-topics that could create a new solution. Up to here, there is not a clear solution. However, it is possible to start to outline some critical steps in order to implement a semantic technology in commercial applications:

a) Implement extractors; transform unstructured information into a formal language.

b) Structure the information into ontologies.

c) Use reasoners to validate the information. A validation assessment is an open question that will require further research.

d) Implement graphic interfaces usable by the final users, where the results of the knowledge processing of the semantic web can be understood easily.

Besides, some tools could be handy in the implementation of the solution:

a) Ontology frameworks: to provide developers with the implementation of a common task in the form of reusable code, therefore reducing work and bugs.

b) Ontology editors: tools for the creation and manipulations of ontologies; they enable the inspection, browsing, codifying, and maintenance task.

c) Ontologies repositories: framework used for storing and querying OWL data.

d) Reasoners: inferences engines that enable advanced queries.

e) Semantic web agents: systems that can integrate data from multiple heterogeneous sources.

Description logic, for query answering, is a very active research area in logic-based knowledge representation and reasoning, has a wide range of applications in knowledge and intensive information systems.

1.1.3 Ontologies

Semantic technologies rely heavily on formal ontologies to structure data for comprehensive and transportable machine understanding. Semantic technologies have established a successful bridge between machines and human beings, using shared knowledge [Geerts-00]. Humans and machines could learn from the same knowledge, ideas, or concepts without requiring the support of scholars. Accomplishing this requires the preparation of the information structure. In order for semantic technologies to work correctly, it is necessary to make information explicit. Thus, one has to add metadata that can support to interpret the original information. Semantic technologies consist of a variety of tools and techniques to work with and manipulate a knowledge container called "ontology."

Ontologies are collections of information and have a taxonomy and a set of inference rules. According to Gruber [Gruber-89], they are an explicit specification of conceptualization. For a long time, they have been used by the Artificial Intelligence Community to share and reuse knowledge between people and machines [Fensel-04].

1.1.4 Description logic

With the introduction of ontology languages, [Cimiano-06] suggests using expressions of description logic language to validate ontologies, becoming description logic in an essential task in creating and populating ontologies in the ontology learning process.

1.2. Treatment of the meaning of the text

A wide range of approaches in NLP does not use logical reasoning to perform their tasks, nor do they take into account the logical phenomena in the documents, even if NLP is moving away from syntax and shallow classification tasks. We are faced with the necessity to understand the meaning of sentences and the implications that can be drawn from them. In this section, we review grammar and logical formalisms to find the combination of syntactic and semantic methods to improve the information retrieval frameworks.

1.2.1 Text analysis

Text analysis in NLP has benefited from recent enhances in the statistical and probabilistic methods such as latent semantic analysis (LSA) [Landauer-13], long short-term memory (LSTM) [Hochreiter-97], Skip-Gram [Mikolov-13a], amongst others [Escobar-18]. New models like bidirectional encoder representations from Transformers (BERT) [Devlin-18] and generative pre-training (GPT-2)[2] made clear the potential of a language model. GPT-2 is a large transformer-based language model with 1.5 billion parameters that can create texts in synthetic form, analyzing and predicting full-text blocks. To have a good performance, this model requires at least 40 GB of tagged information. The language is infinite and evolutive. Even if we have vast data quantities, we need logical and grammatical tools that enhance algorithms that interpret the human language. For this, we review grammar and logical models that can improve statistic-based models. In the first part of this section, we examine some grammar approaches. Finally, we show the results obtained from our experimentation, where we used all the methods involved in this work.

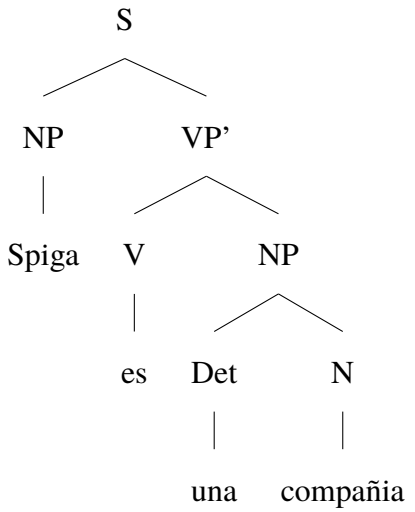
1.2.2 Grammar-based approaches

We require to represent language meaning so that a machine can interpret this representation. Techniques for this representation, such as context-free grammars (CFG), make a simple task, Fig. 1.2. Although CFG is limited to provide a linguistically adequate analysis for some natural language phenomena, grammar formalisms such as CFG can be interesting for computational

[2]GPT-2, Language models are unsupervised multitask learners, Jan. 1 2019, https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

```
S   -> NP VP
VP  -> V NP | V NP PP
PP  -> P NP
V   -> "es" | "contratando"
NP  -> "Spiga" | "Luis"
          | Det N
          | Det N PP
Det -> "una" | "un"
N   -> "compania"
P   -> "en" | "por" | "con"
```

(a) Parsing rules



(b) Forest structure

Fig. 1.2 Parsing rules structure. Example of noun phrase apposition before main verb.

linguists. Recently, [Kallmeyer-10] obtained promising results extending the parsing techniques from CFGs and extending with tree adjoining grammar tree adjoining grammar (TAG), linear context-free rewriting systems (LCFR), context-free grammars (MCFG) and range concatenation (RCG). However, it has been known that the ambiguity problem for context-free grammars is undecidable [Shieber-88]. In this regard, [Brabrand-06] proposes an answer using techniques for statically analyze the ambiguity of context-free grammars. He uses different methods based on local regular approximations and grammar unfolding to extend CFG and adopt diverse formalisms to cope with parsing complexity. He concludes that a statistical analysis of context-free grammar is sufficiently precise and efficient to be valuable in practice. In CFG, grammar methods are defined as a tuple of the form recursive rules used to create strings patterns. Here CFG, G is defined by

$$G = \langle N, T, p, s \rangle \tag{1–2}$$

where N is an alphabet of non-terminals for G, T is an alphabet of the terminal for G, $p \in N$ is the start non-terminals, and $s : N \to P(E^*)$ is the productor function where $E = (T \cup N)$. For instance, the rule defined in Fig. 1.2 when two adjacent noun phrase (np/sn) chunks –the second having a proper noun (np) as head– are found with a verb (verb) immediately to their left, the second noun phrase becomes a child of the first, and the root of the resulting tree is relabeled as (pp/vp). When

TABLE 1.1. ADJECTIVES AND NOUNS WITH HIGHER RESULTS

PoS	***Ancora***		***SenSem***	
	Real	***Simple***	***Real***	***Simple***
ADJ	**94.74**	**97.30**	**91.43**	**91.43**
CONJ	58.82	33.33	55.00	50.00
NOUN	**91.16**	**94.54**	**90.71**	**94.27**
PRON	81.48	92.59	92.59	97.10
ADV	53.85	74.07	83.33	96.15
PREP	71.07	83.05	69.23	80.92
VERB	72.73	96.55	78.26	95.87

the tree-completion task is completed, the tree is straightforwardly transformed into a grammar tree structure. Here, each rule is explicitly tagged by the shallow parser and by the tree-completion step.

When we parse using CFG, we do not know the meaning of its elements, and it depends on grammar rules. However, we are using this lexical approach where we consider that the semantic components of the predicates determine that a verb takes part in a determined diathesis[3] [Pinker-89]. Sometimes parsing serves only to check the correctness of a string; that the string conforms to a given grammar may be all we want to know, e.g., because it confirms the hypothesis the grammar indeed correctly describes specific observed patterns, we have designed for it [Grune-08]. Table 1.1 shows the accuracy from datasets in Spanish, Ancora [Taule-08] and SenSem, the corpora and their single clauses variants are found on those nodes placed near to terminal nodes like determiner (det), noun (noun) or adjective (adj) [Lloberes-10]. Natural languages have a wide range of grammatical structures that are difficult to handle using the simple methods described with CFG. We can extend a feature-based grammar adding sub-categorization, inversion, and auxiliary verbs, unbounded dependency constructions, and cases and gender in Spanish. In an attempt to get more flexibility, we adjust our processing of grammatical categories like *S*, *NP*, and *V*, and substituting with atomic labels. Then we classify into structures where features can get on a series of values. More recently, [Blackburn-05] applies common sense reasoning with episodic logic. Episodic logic is a first-order

[3] Another name for voice.

logic which focuses primarily on time-bounded situations (events and states), rather than the time-insensitive predicates of conventional first-order logic. In another line of work, Rastier [Rastier-96] and subsequently, Tanguy's works [Tanguy-97b] present a model for assigning meaning by semantics traits. Far away from the limitations of logical methods, interpretive semantic analysis is based on semantic fields theory, avoiding the logical models of expressivity and fragmentation generated by logical-based approaches in subsequent works. We will review this theory in chapter 4. In review, we propose to use free grammars as part of the task of meaning extraction from texts without underestimating the complexity of detecting meaning. We know that there are more models involved. This section is a first step. In the following chapters, we will learn how to beneficiate from logical and grammatical validation from semantic context to enhance information extraction and retrieval.

1.2.3 Logical approaches

Spanish parsing does not appear to have support for these component meaning representations. We have followed two fundamental notions in semantics. First, that declarative sentences are true or false in certain situations. Second, that that definite noun phrases and proper nouns refer to things in the world. In IR, we can locate information on an unstructured nature that satisfies an information

TABLE 1.2. PERFOMANCE FOR GRAMMAR AND LOGICAL METHODS

Type	***Method***	***Dataset***	
		Spiga-chat	***SQuad 1.1***
Statistical	MaxEnt	47.10	49.23
Grammatical	CFG	**68.10**	26.41
Logical	FOL	32.12	12.43

need from within extensive collections [Manning-08]. The aim of tackling semantic models is by no means novel in IR. Among others, Montague's work [Partee-76] and [Steedman-01] attempt to analyze semantics about grammar structure and logic entailment.

Montague introduces the notion of universal grammar, where he applies logical methods of formal syntax and semantics to natural language. His natural language semantics approach is based

on formal logic, high order predicate logic, and lambda calculus. Montague argues that natural languages and formal languages are like programming languages and can be treated in the same way. The mechanism for doing Montague's Semantics involves, in a fundamental way, a recursive syntactic rule called WFFS (well-formed formulas). They are starting with these smallest, primitive elements and specifying how units of various categories can be combined to form large units. The semantics task is to assign interpretations to the smallest units and then give rules that determine the interpretation of larger units based on the interpretation of their parts. A vital feature of this approach is that the complete analysis should be the same in the syntax and the semantics, e.g., in "*Spiga está contratando*," and "*Algunas empresas están contratando*," "*Spiga*" is an expression of the category Proper Name. Its denotation is an individual represented in logic by "*Spiga*." The Intransitive Verb *contratando* denotes a set of companies it is represented by the predicate symbol "*hiring.*" The Common Noun "*company*," which denotes a set, represented by the company. The determiner denotation is

$$\lambda P \lambda Q \exists x[P(x) \rightarrow Q(x)] \tag{1–3}$$

its explanation defined by tree rules; (a) we take as input a proper name and produce a noun phrase. (b) we use as an input a noun phrase and an intransitive verb and yields as output a sentence: from "*Spiga*" and "*contratando*" it produces "*Spiga está contratando.*" (c) Furthermore, third, that takes as inputs a determiner and a common noun and yields a noun phrase: from "*Algunas*" and "*Spiga*" it produces "*algunas empresas*." The example given with the last rule helps us to understand the formula for every set: that denotes a relation between properties A and B which holds in case every A has property B. The next step is complex. It applies the rule for combining a noun phrase and an intransitive verb to the last result (1–3), producing "*Algunas empresas están contratando.*" The output of the semantic rule is

$$\lambda Q \exists x[empresas(x) \rightarrow Q(x)](contratando) \tag{1–4}$$

$$\exists x[empresas(x) \rightarrow contratando(x)] \tag{1–5}$$

which is the traditional logical representation of "*Algunas empresas están contratando.*" While

interpretive semantics maintains the distinction between syntactic rules, but not post any systematic relation between them. Table 1.2 shows the requirement that semantic interpretation rules correspond structurally to the syntactic rules that can put very strong constraints on possible syntactic analyses. This point is worth to be emphasized because Montague offers virtually no constraints on syntactic rules themselves; it is only in the connection between syntax and semantics that the grammar is constrained, but that constraint is strong enough that is a serious open question whether natural languages can be so described.

1.2.4 Propositional logic

Logical language makes reasoning formally explicit. In propositional logic, we have propositions, usually denoted by capital letters, and connectives between these propositions. We develop logical representations of a sentence that formally obtain the truth-conditions of propositional logic that allows us to represent parts of the linguistic structure that correspond to specific connectivity. Logics gives us an essential tool for performing inference. The sentence interpretation of a logical language is a simplified version of the world. The propositional logic required to set boolean values to simplify is in (1–3). We can do that with positional symbols, e.g., (a) "*Nuestra compañía está al norte del país.*" Therefore, "*El país no está al norte de nuestra compañía.*" Translating sentences into propositional logic can represent only atomic sentences. We cannot understand their internal structure. In effect, there is nothing of logical concern in dealing with nuclear sentences into predicates, objects, and subjects. However, this seems incorrect: if we want to normalize parameters, we must be fit to view inside. We move to propositional logic to the more expressive FOL. Not all of natural language semantics can be expressed in FOL. However, it is the correct alternative for computational semantics because it is expressive enough to represent a good deal. There are systems available off the shelf for carrying out automated inference in FOL, the syntax FOL boolean operators of propositional logic, and the symbols used as predicates do not have intrinsic meaning.

A. First-order logic

The FOL equivalent of a context of use is variable binding, which is a map of individual variables to entities in a given domain. We use the principle to build a logic form where the meaning a function of the meanings of the parts and the form they are syntactically mixed. The theory of syntactic analysis gives the semantically relevant parts of a complex expression. We will take expressions that

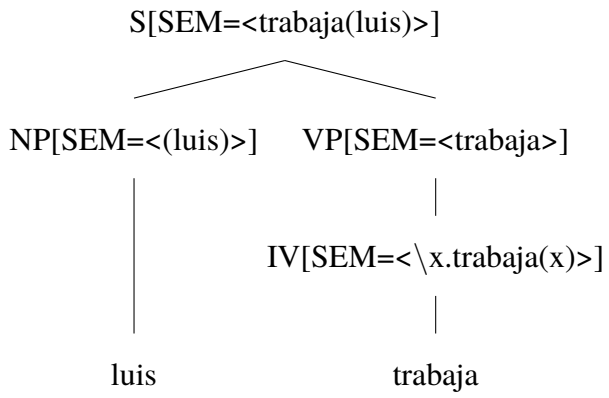


Fig. 1.3 Semantic compositional analysis representation.

are parsed against CFG. However, this is not required by the principle of compositionality. We aim to combine the development of semantic representation in a way that the parsing process can be more efficient. Fig. 1.3 illustrates the first approach to the class of analysis we would like to produce, the SEM value at the origin node displays a semantic representation for the whole sentence. In contrast, the SEM values at lower nodes show semantic representations for constituents of the sentence. Since the values of SEM have to be treated in special manner, they are distinguished from other feature values by being enclosed in angle brackets. We will assign semantic representations to lexical nodes, and then compose the semantic representations for each phrase from those of its child nodes. However, in the present situation, we will use function application rather than string concatenation as the mode of composition. To be more specific, suppose we have an NP and VP constituents with appropriate values for their SEM nodes; then, the SEM value of an S is handled by a rule like (1–4). Observe that in the case where the value of SEM is a variable, we omit the angle brackets.

$$S[SEM =<?vp(?np) >] \rightarrow NP[SEM =?np]VP[SEM =?vp] \qquad (1\text{–}6)$$

Formula (1–6) tells us that given some SEM value *?np* for the subject *NP* and some SEM value *?vp* for the *VP*, the SEM value of the S parent is constructed by applying *?vp* as a function expression to *?np*. From this, we can conclude that *?vp* has to denote a function that has the denotation of *?np* in its domain. Formula (1–6) is an example of building semantics using the principle of compositionality. To complete the grammar is very straightforward; all we require are the rules shown below.

$$VP[SEM =?v] \rightarrow IV[SEM =?v] \tag{1–7}$$

$$NP[SEM =< \text{'Luis'} >] \rightarrow \text{'Luis'} \tag{1–8}$$

$$IV[SEM = (\text{trabaja}(x))] \rightarrow \text{'Luis'} \tag{1–9}$$

the VP rule (1–7) says that the parent's semantics is the same as the head child's semantics. The two lexical rules provide non-logical constants to serve as the semantic values of "*Luis*" and "*trabaja*" respectively. We use *set theory* that provides us with a tool for combining expressions of first-order logic as we assemble a meaning representation for a sentence. Set theory is a helpful method of specifying properties P of words, e.g., $w|w \wedge P(w)$, which we glossed as the set of all w such that w is an element of V (the vocabulary) and w has property P. To be extremely valuable, we add something to first-order logic that will achieve the same effect. We do this with the λ operator counterpart, e.g., $\lambda w(V(w) \wedge P(w))$. Since we are not trying to do set theory here, we just treat V as a unary predicate. λ is a binding operator we can bind the variable x with the λ operator, as $\lambda x(trabaja(x) \wedge estudia(x))$. We have a particular name for the result of binding the variables in an expression: λ-abstraction.

Determining the number of nouns is a crucial issue. e.g., "*Un estudiante trabaja.*" A logical form of this will be $\exists x(estudiante(x) \wedge trabaja(x))$. Another thing that we need to handle is to deal with sentences containing transitive verbs, e.g., "*Luis logra sus metas,*" we expected to have $\forall x \exists y(metas(x) \wedge y = luis \leftrightarrow cumple(y, x))$. Here, we can use λ–abstraction to obtain this result. A limitation of this syntax-driven is the scope ambiguity. Semantic representation is highly related to the syntactic analysis. In this case, quantifiers in the semantics have a relative scope. Consequently, a sentence like "*Cada trabajador alcanza las metas*." repeated here, will always be translated as (1–10), not (1–11).

$$\forall x(\text{trabajador}(x) \rightarrow \exists y\ (\text{metas}(y)\ \wedge\ \text{alcanza}(x, y))) \tag{1–10}$$

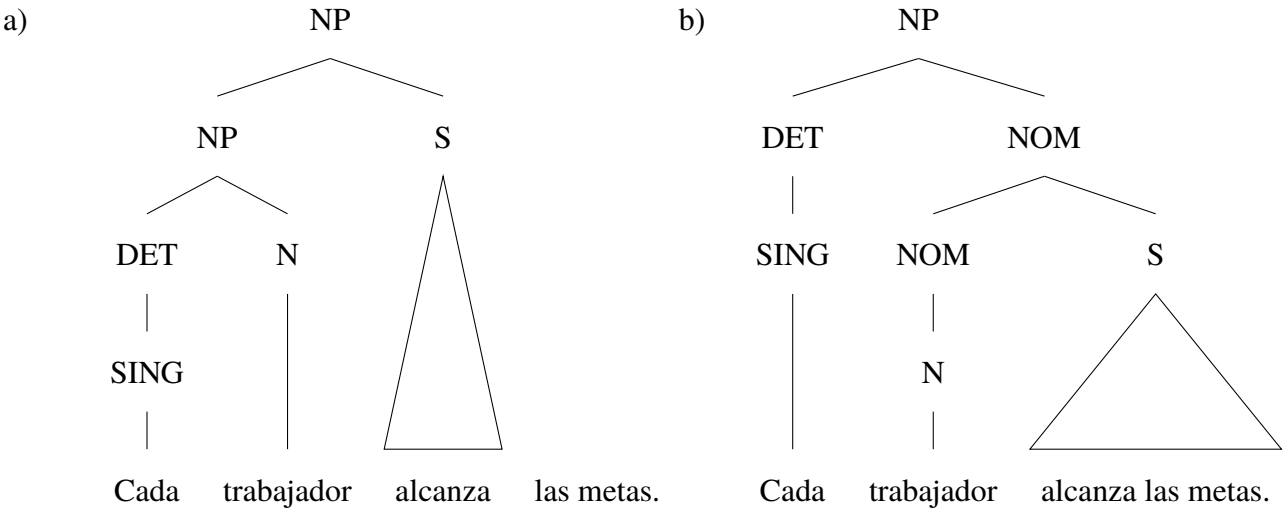


Fig. 1.4 Quantifier scoping. The structure (b) provide a direct basis for the semantic interpretation in a way that (a) cannot.

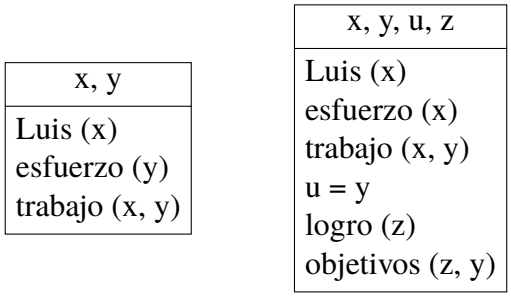


Fig. 1.5 An example of the discourse representation structure.

$$\exists y(\text{metas}(x) \rightarrow \forall y(\text{trabajador}(y) \ \wedge \ \text{alcanza}(x, y))) \tag{1–11}$$

There are approaches to dealing with scope ambiguity. Cooper [Keller-88] storage method proposed a pair consisting of a core semantic representation plus a list of binding operators. S–Retrieval, and Address [Uwe-88]. Fig.1.4 depicts how the two readings of (1–10).

1.2.5 Paragraph treatment

The last topic for this section deals with paragraph treatment when we are analyzing sentences. Sequential sentences form a paragraph, their interpretation depends on reminding the elements of the previous elements. e.g., an anaphoric pronoun, such as *he*, *she*, and *it*. FOL is limited to

single sentences. Nevertheless, we can extend the scope over two or more sentences. e.g., "*Luis se esfuerza en su trabajo. Él logra sus objetivos.*" $\exists x \forall y(\textit{trabajo}(x) \wedge \textit{esfuerzo}(\text{luis}, x) \wedge \textit{objetivo}(y) \rightarrow \textit{logra}(\textit{luis}, y))$. That is, the *N* a "*trabajo*" acts like a quantifier that binds the "*sus*" in the second sentence. The discourse representation theory (DRT) [Kamp-85] was developed with the specific goal of providing a means for handling this and other semantic phenomena, which seem to be characteristic of discourse. Discourse representation structure (DRS) presents the meaning of discourse in terms of a list of discourse referents and a list of conditions. The discourse referents are the things under discussion in the discourse, and they correspond to the individual variables of FOL. The DRS conditions apply to those discourse referents and correspond to open atomic formulas of FOL. Fig. 1.5 illustrates how DRS for the first sentence in (a) is augmented to become a DRS for both sentences. The discourse representation structures; the DRS on the left-hand side represents the result of processing the first sentence in the discourse, while the DRS on the right-hand side shows the effect of transforming the second sentence and integrating its content.

1.2.6 Experiments

To evaluate our method's effectiveness and generality, we tested it on one dataset of chatbot dialogs, characterized by open questions. There are several tests for QAS, like TREC[4] , SemEval-2017 Task 3 questions[5] and QALD[6]. The most state-of-the-art method is SQuaD[7], but it only works with the English language. The test for our Spanish approach creates a dataset named SpigaChat[8] with 10,000 dialogues taken from conversations of other chatbots. Our experiments consist of increasing the text analysis's complexity and applying statistical methods in combination with logical methods. We use maximum entropy (MaxEnt) as the statistic baseline [Escobar-18]. Afterward, we perform the grammatical analysis that we review in section 4 with CFG. Finally, we validated information retrieval (both in the answer and in the question) using lambda expressions and FOL inferences. We obtained a sufficient analysis level to generate primitive ontologies, class, relation, class; that is, we can identify conceptual elements and their relationships. We present results in Table 1.2,

[4]TREC, Question Answering Collections, Apr. 2, 2019, https://trec.nist.gov/data/qa.html.

[5]SemEval, Multilingual and Cross-lingual Semantic Word Similarity, Apr. 2, 2019, http: //alt.qcri.org/semeval2017/task2.

[6]QALD, Question Answering over Linked Data, Apr. 2, 2019, http://qald.aksw.org.

[7]SQuAD, The Stanford Question Answering Dataset May. 26 2019, https://rajpurkar.github.io/SQuAD-explorer/.

[8]Spiga.ai, Spiga chatbot Question Answering Dataset Jan. 1 2020, https://spigaai.github.io/spigachatai/.

alongside prior work and strong baselines. The method that has better results is MaxEnt [Nigam-99] with 70.1%, with more information labeled, the better the obtained result, except when questions arise that should not be answered, e.g., "*¿Qué fue primero, el huevo o la gallina,?*" The algorithm must be strict, with the weight of the syntactic elements of the statement, or it will always answer. However, the cost of training is exponential. There is an improvement with CFG, 72.1% (+2), but spelling mistakes make the initial tasks of part-of-speech tagging (POS) and the execution of the rules confusing. FOL did not bring significant results 72.3% (+0.1); the logical validation at this level is still affected by syntactic ambiguities, given the lack of meaning in the elements.

1.2.7 Conclusion

We have explored the logical formalism for performing inferences directly on the syntax surface form text. This formalism has the advantage of giving us a logic notation to verify our semantic analysis but also has some possible faults since we are still far from the automatic generation of logical syllogisms. In the following chapters, we advance our study, focusing on semantic interpretation and question answering tasks.

2. Transformations

This chapter demonstrates how to transform information from invoice documents to semantic models to implement ontology modeling. We propose a solution to ontology modeling issues when mapping a document that follows some extensible markup language (XML) schema to an ontology under the OWL standard. We provide new interpretations of the XML terms in the context of OWL so that the XML schema definition structures can be mapped into more complex OWL structures. Finally, we developed a tool to test the proposed information extraction strategies.

2.1. Introduction

The latest changes in Mexico's tax management platform has opened some opportunities to overcome the technological lag in this country. The amount of semantic information made available by the Mexican Government[1] for open use will allow the introduction of semantic tools that have already been implemented in other countries[2,3]. The internet digital fiscal receipt (CFDI from its name in Spanish) is the current model of electronic invoice valid in Mexico since January 2011. This type of receipt, which uses standards regulated by the government fiscal agency in Mexico, is constituted as a digital document in XML[4] that has the following characteristics:

a) Integrity: the information contained in a internet digital fiscal receipt, from its name in Spanish (CFDI) cannot be manipulated nor modified without being detected.
b) Authenticity: the identity of the generator of the receipt can be verified through its Digital Certified Seal.
c) Unique: each and every CFDI has attached a registered identifier given by an approved certification supplier that transforms the receipt into the link between its addressee and the government.
d) Verifiable: the person emitting the CFDI could not deny having emitted it. CFDI is obligatory

[1]Mexican Government, Estrategia Digital Nacional (EDN), May 29, 2015, http://www.presidencia.gob.mx/edn.
[2]Data.gov, Developers, Semantic Web, Jul. 29. 2017, https://www.data.gov/developers/semantic-web.
[3]Datos.gob.es, Ontology, Jul. 29, 2017, http://datos.gob.es/es/talk-tags/ontology.
[4]W3C. Extensible Markup Language, Jul. 29, 2017, http://www.w3.org/XML.

to be used in every commercial or business operation.

The CFDI brings opportunities to the companies that develop commercial applications by facilitating, interacting, and accessing semi-structured information, which makes it possible to develop systems to manage the commercial knowledge, information search engines, electronic commerce platforms, information management agents, and knowledge managers, to mention but a few. The bottom line will be in favor of the final users by enabling them to get a better understanding of their businesses. However, to reach this aim, it is required to transform the information into knowledge. In the late 90s, Tim Berners-Lee was concerned about the fact that information itself on the web was not enough to make the computers understand the knowledge that was being generated. Even though the HTML documents can be linked through hyperlinks, they are isolated documents, which makes it complex to share information [Berners-Lee-01]. This may be a gap that causes severe problems in accessing and processing the available information; especially in searching for information, presenting information, and electronic commerce [Fensel-05]. One might think that the migration from CFDI to semantic documents could facilitate the creation of complex tools that lead to a deeper understanding of the information as knowledge, and the regular user (without technical skills) would have a tool straightforward to use. However, there are some gaps to be solved before using these semantic documents. It is necessary to model knowledge accurately. The CFDI are structures that are nested, but additional information (implied) is required to transform them into a semantic model successfully. In this work, the information is differentiated according to its abstraction degree. That is, there are some words of popular use that are commonly found in dictionaries, encyclopedias, and that are taken as concise concepts or ideas; on the other hand, there are some other words such as names of persons, products, streets, among others, that are considered as assertions of the concepts. The latter are also known as individuals. The CFDI, as a semi-structured document, is composed of both types of information. In description logic, this is defined as terminological component (TBox) for naming the concepts and assertion component (ABox) for naming the instances. For example,

$$Man \equiv Person \sqcap Male \tag{2–1}$$

where a male can be defined as a male person by writing this declaration. Similarly,

$$Male \sqcap Person(\text{PABLO}) \tag{2–2}$$

states that individual PABLO is a male person. Given the above definition of man, one can derive from this assertion that PABLO is an instance of the concept Man [Baader-10]. In the CFDI, it is nonfrequent that the TBoxes change unless there is a new rule that demands to modify the current elements this happens when the congressmen reach consensus on the need for new policies on the fiscal processes. On the other hand, the ABoxes suffer alterations that are not necessarily related to the TBoxes. For instance, an invoice has descriptions of products using natural language always, e.g., *"Aspirin tablets 10 mg. for Infants."* If this text is read, it can be observed that the description is about some medicine, the amount of active formula, and the recommended user (children). Modeling of these types of chains is what makes ontology modeling complex, especially when the modeling pattern is being created dynamically. Due to the complexity of analysis, this document excludes the transformations of this kind of information and takes them as informative chains. In this report, some of the strategies that are taken to transform the information of the CFDI documents are presented, describing the considerations and the different components used to implement and reach a solution. The final aim is to build a software tool that can extract and transform the information of several CFDI into an ontology that follows the OWL standard. This ontology will be used as the base for a QAS that uses the knowledge represented to provide some decision-makers with insights about its business in a more user-friendly way.

2.2. Invoice documents in XML

The XML used to describe the CFDI is composed of information about the purchase done by a client. The data is clustered around the basic concepts of the sale, e.g., the *"Client"* element contains information from the customer, such as the address. The *"Provider"* is referred to as the one who closes the sales, or offers the service, and the *"Concept"* element is a list where a complete description of the products and services of the sale is included. An example of a CFDI document is shown in Fig. 2.1. The challenge comes because the structure of XML schema often contains implicit assumptions about taxonomy and relationships. However, for the intended meaning (e.g.,

```
<?xml version="1.0" encoding="utf-8"?>
<cfdi:Comprobante
    xsi:schemaLocation =
    "http://www.sat.gob.mx/cfd/3  http://www.sat.gob.mx/./cfdv32.xsd" folio="20474"
    xmlns:cfdi="http://www.sat.gob.mx/cfd/3" xmlns:xsi="http://www.w3.org/2001/XMLSchema...">
    <cfdi:Emisor>...</cfdi:Emisor>
    <cfdi:Receptor>...</cfdi:Receptor>
    <cfdi:Conceptos>
        <cfdi:Concepto unidad="CAPSULAS" importe="244"  cantidad="1.0"
        descripcion="VIBRAMICINA 100MG" valorUnitario="244.00" />
        <cfdi:Concepto unidad="BOTELLA" importe="137.93" cantidad="1.0"
        descripcion="CLORUTO 500M" valorUnitario="137.93" />
        <cfdi:Concepto unidad="TABLETAS" importe="84.5" cantidad="1.0"
        descripcion="SEDEPRON 250MG 10" valorUnitario="84.50" />
    </cfdi:Conceptos>
</cfdi:Comprobante>
```

Fig. 2.1 CFDI document: electronic invoice with levels of nesting.

the semantics) to work, it is required that the resources be explicit in a manner that are understood by computers [Baader-10]. This makes it challenging to implement a wholly general and automated mechanism to transform XML schemas into OWLs ontologies. XML provides a language for describing the structure of information to support automated processing [Hitzler-11]. The XML schema definition XML schema definition (XSD) language contains type and element definitions that describe characteristics of well-formed elements and attributes of XML documents[5]. However, the XSD language does not express semantics[6], creating difficulties for semantic web technologies. The resource description framework (RDF)[7] standard was created to represent information structures. Its syntax is defined by data structures that represent graphs that are sets of *subject-predicate-object* triplets, where the elements may be uniform resource identifiers (URIs), blank nodes, or data typed literals. The main objective of the RDF is to express descriptions of resources [Fensel-04]. On the other hand, the OWL[8] specification was developed to enable semantics. It extends RDF with terminology to express ontologies using DL, a decidable fragment of first-order logic [Baader-10]. OWL DL ontologies are a subset of OWL that satisfy the expressive requirements of description logic.

[5]XML Schema, Mar. 1, 2017, https://www.w3.org/TR/2004/REC-xmlschema-0-20041028.

[6]T. Berners-Lee. Why RDF Model is Different from the XML Model, Jul. 29, 2017, https://www.w3.org/DesignIssues/RDF-XML.html.

[7]RDF Schema, Jul. 29, 2017, https://www.w3.org/TR/2014/REC-rdf-schema-20140225.

[8]Web Ontology Language Semantics and Abstract Syntax. Jul. 29, 2017. https://www.w3.org/TR/owl-semantics.

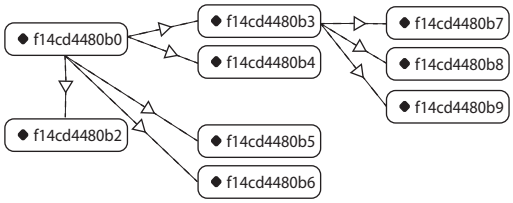


Fig. 2.2 XML nodes linked to OWL elements: individual relation results (protégé).

2.3. Transforming a CFDI to an OWL

An invoice results from a purchase made by a client. The CFDI represents this invoice, and it is composed of an issuer and an addressee, the description of the sold items, and a description of the taxes applied in the purchase. The CFDI, just as all the XML, is handled by an XSD schema, where it is defined which fields, lengths, and types of fields are required. The element *"Concept"* is the root and contains the other elements; it might be nested depending on other concepts. Just like the CFDI, the ontology is described in an XML format; this facilitates the transformation of the documents into ontologies. Both use schemes and namespaces. The CFDI uses the schemas from Secretaría de Hacienda y Crédito Público (SHCP). Meanwhile, the OWL uses those from world wide web committee (W3C) and semantics. However, this may have some complications while analyzing the information in a semantic form, e.g., a logic reasoner could not use a CFDI directly, since, in a first instance, it needs that the information contained in the document be explicit, both concepts and relationships. Fig. 2.2 shows what is expected: every XML node is linked with an OWL element, and sparql protocol and RDF query language (SPARQL) is used in a simple way to test it.

2.3.1 Mapping strategy

There are some approaches to transform XML into OWL/RDF; most of them are based on linear mappings. Redefer [Garcia-05] is a framework that demonstrates how to map transformations, and can work in environments such as digital right management (DRM); on the other hand, Ontomalizer[9] is a tool to improve metadata enrichment and semantic processing for biomedical documents. There

[9]Ontmalizer, A Tool that Performs Comprehensive Transformations of XML Schemas (XSD) and XML Data to RDF/OWL Automatically, Jul. 29, 2017, https://github.com/srdc/ontmalizer.

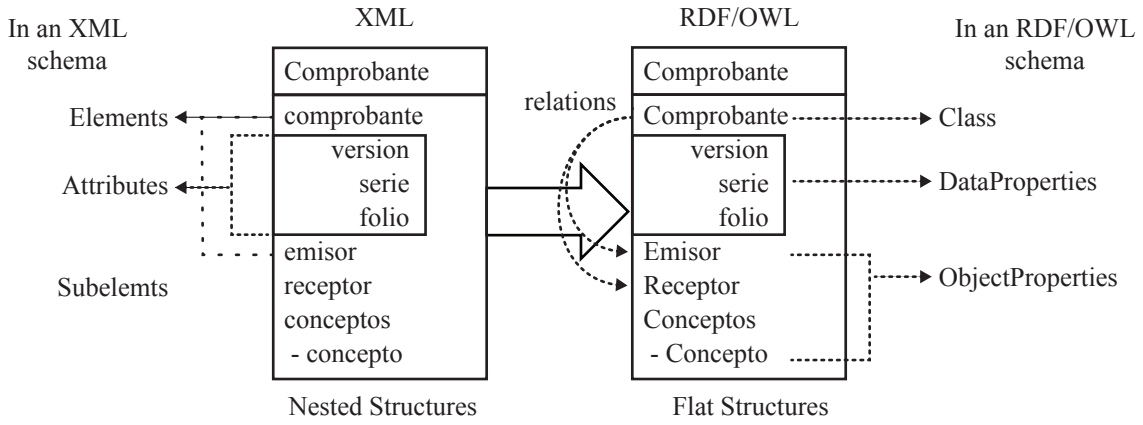


Fig. 2.3 Semantic transformation: XML is parsing to RDF structure.

are other open domain frameworks that can also be used in mapping transformations: OpenRefine[10] is a tool to transform data from one format into another, including XML or RDF, and TopBraid[11] that recently included the ability to convert XSDs and associate XML files to RDF and OWL. Ferdinand's work [Ferdinand-04] was one of the first approaches that proposed the use of conversion rules to deal with the mapping of the concepts in the CFDI documents. Its use could be applied to just one document as well as to groups of XML documents creating a more complex ontology. The proposal described in this work considers some topics as the one from mapping concepts of reasoning support for web engineering, and also Ferdinand's work ,for mapping. However, many of the proposals here stated are quite new. The detention of conceptual elements, not only the concepts mapping but also their recognition, is also proposed in new ontological modeling using a horizontal structure that gives more consistency to the nodes or relations in the ontology. The synthetic URIs were also introduced to avoid the loss of non-schematic information. These are features that permitted that the number of concepts and relations of the resulting ontologies were improved. Fig. 2.3 shows the complete panorama of this transformation. It is assumed that the schema of the files of an XML document can be interpreted as conceptual ontologies, that is, an XSD file should be transformed into an OWL file without any complication in a linear way. It happens the same when transforming XML files into RDF files. However, this only works in the theoretical field. The first difficulty comes up when the document contains several namespaces (NS). An iterative process is implemented to

[10]OpenRefine, A Free, Open Source, Power Tool for Working with Messy, Jul. 29, 2017, http://www.openrefine.org.
[11]TopBraid. Visual Modeling Environment, Jul. 29, 2017, http://www.topquadrant.com/topbraid.

transform every single NS found. However, when the complex schema contains a large number of references, the recursive process can be delayed or collapsed. In order to avoid this problem, a maximum level of iterations is defined, and synthetic namespaces are created that help to define the scope of the transformation of the referred NS. In this way, the recursiveness levels are limited, and the time of the transformation is reduced by improving the transformed NS. An OWL file cache of the transformations is also started; this accelerates the process, mainly when the information uses references that are public domain, such as encyclopedias and geographic references, among others. The proposed steps for mapping an ontology are:

s1. Create a group file with the transformation rules based on standard OWL.

s2. Obtain the NS from the XML and transform them into an ontology.

s3. Define a recursive level.

s4. Determine the policies and rules for the generation of synthetic NS.

s5. Measure the quality level of the generated ontologies.

s6 Execute a SPARQL and DL queries to validate the ontology.

2.3.2 XSLT Transformations

In a first approach, the extensible stylesheet language transformations (XSLT)[12] technology was used to provide support to several forms of transforming. This was made through transformation rules, many of them using the XQuery[13] and XPath[14] data model. The OWL 1.1 standard[15] was chosen. Every single father element was taken as an OWL class. This happened as well to every attribute, where each one was joined with a class through the properties.

2.3.3 Recurrence and mapping complexity

Recurrence works in a linear sense; however, when the cycles are not homogeneous, that is, when a son loop uses data from a father loop, the complexity starts to be high in the script, and this could be even more complicated when the levels of nesting are higher than three, due to the exponential growing. Fig. 2.1 shows an electronic invoice that has about six levels of nesting. The

[12]W3C, Transformation W3C, Jul. 2, 2017, https://www.w3.org/standards/xml/transformation.
[13]W3C, An XML Query Language W3C, Jul. 2, 2020, https://www.w3.org/TR/2017/REC-xquery-31-20170321/.
[14]W3C, XML Path Language, Jul. 2, 2020, https://www.w3.org/TR/2017/REC-xpath-31-20170321/.
[15]W3C. OWL Web Ontology Language Overview, Jul. 2, 2017, http://www.w3.org/TR/owl-features.

transformations using XSLT worked well on simple XML that had a few elements and low diversity in the references; however, the maintenance of the XSLT models is complicated, especially when the diversity of the structures becomes large. This requires the development of a new model where the adaptation of the rules with the number of the used structures is dynamic. A hierarchical structure processor based on XML formats was developed. It was simple, though, since many developed components deal with these tasks. Perhaps, their only constraints could be related to their over-usage when the elements have many internal cycles.

2.3.4 Ontological looping

One feature of the ontologies is their ability to be shared and related [Fensel-05]. When a document is transformed, it could be necessary to go beyond the limits of the document to complete the models. This can lead to search for other sources and to process them; however, this could be expensive for the transformer, especially if the resources are not available at the moment.

2.3.5 Duplicity in the XML elements

One of the logics conflicts that are more common occurs when the information that is contained in XML elements is repetitive. For example, in the invoicing of a store, the emitted invoices always have the same information regarding the seller; if these cases were omitted, many identical instances would be generated from the same store that semantically would be treated as different. By incrementing the number of transformed invoices, it makes sense that new assertions make the graph grow. Some elements of the invoices contain information that begins to be repetitive, e.g., when there are several purchases from the same customer, the name, ID, and the address could have been previously registered, and there would be two or more similar nodes in the same graph unlinked. The tax administration service (SAT) schema does not define these types of issues. In other words, they do not mention anything about unrepeatability and uniqueness.

Unrepeatable and Linkable. To identify this node, an additional complement to the SAT XSD is defined, where the node specification can be defined as unrepeatable. This enables a *hash* code with the node and its attributes, ensuring consistency between the links from different CFDIs. In this way, there is a graph distributed with more links and quality, and it avoids the unlinked vertical

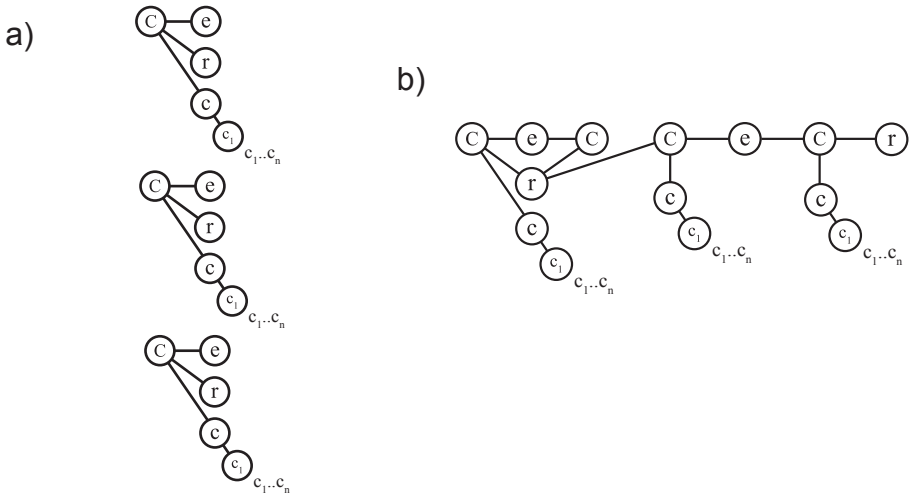


Fig. 2.4 Ontological model: a) vertical tendency, b) horizontal tendency.

growth. If inferences are applied to linked elements, links could be enhanced. Although this task is beyond the scope of this work, it is possible to verify that even with the resulting relationships between entities, reduce the complexity of the queries obtaining better results.

2.3.6 Improving the ontological model

Up to this point, the ontological model has been used just in the individual transformations of the CFDIs. However, as more documents are added to the ontology, it tends to take a vertical shape. Fig. 2.4 (a) shows this effect. This has negative consequences, mainly due to the queries becoming more complicated when using regex patterns in their statements. When the number of triplets, links among concepts, is smaller, the effectiveness of the inference is reduced, and the final results are significantly weakened.

To avoid this, some adjustments are made: a conceptual validation is implemented by applying the *hash* algorithm, which avoids repetition and helps to coagulate similar concepts. However, this is not a definitive solution. There are some attributes within the elements of the XML that cannot be hashables, such as dates, addresses, balances, and the like. Some of these attributes could be handled just through heuristics. Fig. 2.4 (b) shows the result with the adjustments included. It can be noted that there is more cohesion of the nodes and less redundancy of the information.

2.3.7 OWL Frameworks

Initially, Jena[16] framework was selected because it is the open-source community option, and recently, Apache foundation adopted it once HP[17] left it open. The architecture of this framework is based on RDF models that, from time to time, are modified to make them adequate to new OWL standards; however, the updating has not been completed. According to its documentation, they are not fully compatible with the last version of OWL 2 but supports only some characteristics. The main issues come up when the modeling of the individuals of a class are started. The Jena models, though they are configured to use OWL DL, do not create the model according to the OWL 2. They recur to the clustering of RDF kind classes but they do not use the *owl:NamedIndividual* labels. This may cause performance failures, especially when the ontologies start growing. Owlapi [Horridge-11] is one of the most robust frameworks nowadays. This model treats an ontology as a set of axioms rather than as a set of triplets. It follows the guidelines of the OWL specifications, and due to this, it is enough to write OWL 2 ontologies.

2.3.8 Conceptualization of the information

Another challenge found consists of how to determine the conceptualization level that the transformed document should reach. It was determined that the conceptualization level reflects the logical processing capacity that the document could have. A description of a strategy to deal with the conceptualization of the XML files is in the following section.

2.4. Ontoparser

To validate the previously described methodology, the Ontoparser software is developed Fig. 2.5, which allows managing the transformations of XML schemas and XML data to RDF/OWL automatically. Ontoparser works in a web application where the clients upload CFDI of the purchases/sales that they have made. The user can create a repository to save the information of the invoices that will be transformed into graphs. It is essential to show the user the size of resource/space that is available. Ontoparser takes as a measurement unit the number of triplets that are in a repository. It can be

[16] Apache Jena. (2017, Jul. 29). Apache Jena - Home [Online]. Available: http://jena.apache.org.
[17] HP. (2017, Jul. 29). Hewlett-Packard Company [Online]. Available: http://www8.hp.com.

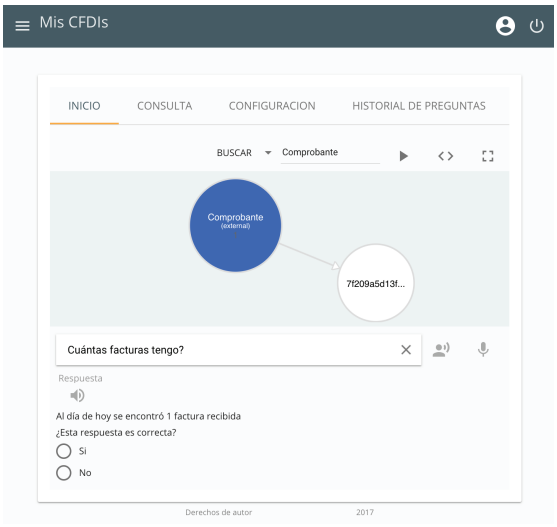


Fig. 2.5 Ontoparser: resultant graph from a user query.

stated that the higher the number of triplets are in a graph, the more processing resources are needed, and the more expensive the service becomes. Once the reception of the XML or zip file is finished, if there are several XMLs, a process in the background is executed to make the transformation task. The user can know how gradually the graph is processing the documents separately. The task does not need to be finalized for the user to make any consultation about the documents that are being uploaded; it might happen that some information might be omitted because it is still not processed. In the current version, it is not possible to delete or update a CFDI individually; it is only possible to delete the whole graph.

Ontoparser uses different storages to save the ontologies. Fuseki TDB[18] is used by default to store the information of a standard user. It has good performance and can hold a large number of datasets. However, this is a tool to be used in safe zones without substantial security restrictions. However, if the information is sensitive and some security standards are required, Ontoparser is ready to use storages like Virtuoso[19], Allegro Graph[20] and Oracle Graph 12c[21], that offer these characteristics, Fig. 2.6 shows a resulting graph from a QUERY in natural language. A user asks about how many invoices have been received today, Ontoparser can even listen to the voice of the user. Ontoparser shows a graph with the information attached to a short answer in audio. All the

[18] Apache Jena TDB, Aug. 28, 2017, https://jena.apache.org/documentation/tdb.
[19] OpenLink Virtuoso, Aug. 28, 2017, https://virtuoso.openlinksw.com.
[20] Franz Inc. AllegroGraph - Semantic Graph Database, Aug. 28, 2017, https://franz.com/agraph/allegrograph.
[21] Oracle 12c Spatial and Graph, Aug. 28, 2017, https://www.oracle.com/database/spatial.

```
/* Sparql name:getFrequentItems */
1 PREFIX cfdi: <http://www.sat.gob.mx/cfd/3#>
2 SELECT ?descripcion(COUNT(?descripcion)AS ?Nit)
3 WHERE //
4  { ?concepto cfdi:Descripcion ?descripcion }
5 GROUP BY ?descripcion
6 ORDER BY DESC(?Nit)
7 LIMIT 5

/* Sparql name:getExpensesLastInvoice */
1 PREFIX cfdi: <http://www.sat.gob.mx/cfd/3#>
2 SELECT ?monto
3 WHERE
4  {?Comprobante cfdi:Total ?monto;
5      cfdi:Fecha ?ultimafecha
6     {SELECT(MAX(?fechaFactura) AS ?ultimafecha)
7      WHERE
8       {?Comprobante cfdi:Fecha ?fechaFactura}
9     }
10  }

/* Sparql name:getTotalAmountInvoices */
1 PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
2 PREFIX cfdi: <http://www.sat.gob.mx/cfd/3#>
3 SELECT (SUM(xsd:float(?importe)) AS ?total)
4 WHERE
5   { ?Comprobante cfdi:Total ?importe }
```

Fig. 2.6 Three different SQPARL queries for testing the new invoice ontology.

APIs can be called from the REST services. There are two primary services to transform an XML document into an RDF, XML to RDF, and XSD to OWL. To transform an XML into RDF, the XML must have well-defined schemes that are used in every node, attribute, or value in the body of the structure. However, if an XML does not have the schemas well defined, the references are created synthetically, mainly with the *knowitive.com* namespace.

2.4.1 From RDF/OWL-DL outputs to XML (XML to RDF/OWL)

It is possible to transform more than one XML if they are concatenated (using ";") with the URIs in the entry string, the OWL output will hold all the information of the URIs received. It is also possible to choose the notation of the OWL output, by default RDF/XML is set, even though some other formats are possible to be chosen, such as RDF/XML–ABBREV, N–TRIPLE or N3. In the last version, the OWL format [Lohmann-16] was added. VOWL[22] provides a JSON notation that facilitates the rendering of the ontology in a GRAPH. The outcome of the transformation can be either a URIs type (that means the file is set/written in storage and then is assigned a URIs) or a body type where the service request contains the complete OWL file. Additionally, the ontological framework used for the transformation can be chosen, Jena or Owlapi.

[22]VOWL, Visual Notation for OWL Ontologies, Jan. 13, 2021, http://vowl.visualdataweb.org/.

2.4.2 From XSD to OWL

This function is similar to that one previously described, but its objective is just to transform a schema from XML into an ontology or TBox. The XMLtoRDF parser module is called internally when in the XML file, a reference to a schema is found. It uses the same parameters as those for the previous service.

2.4.3 Outcome, tests and results

TABLE 2.1. ONTOPARSER PERFORMAMCE RESULTS

Outcome	CFDIs	Nodes	Triplets
Simple	1	38	543
Complex	10	1,000	5,266

The tests are made with different sizes of XMLs. We arbitrarily classified them as Simple, when they are no longer than 40 nodes, and Complex, when they are longer than 40 nodes, or when they use more than one XML (in this case, the nodes went up to 1000). The execution transformation time, using a virtual instance with 1 CPU Intel Xeon 3.3 GHz and 1 GB memory, is not so long to consider it an issue. In any case, the tests are finalized excepting when the XSD schemes are not available, in which case it is necessary to implement a cache. By doing so, it is possible to reduce the transformation time to a few seconds. It was also found that there is not much difference in the performance when using Jena[23] or Owlapi; however, as explained above, the model structures are entirely different. It was decided just to evaluate the outcome obtained with Owlapi since the version of the OWL 1 that Jena uses does not deal with individuals explicitly. The first necessary adjustment was for the Object Properties that do not have a double way. That is, the one way linked between A and B classes was changed to double way to have consistency, mainly in the queries of SPARQL, as illustrated in Fig.2.6. Table 2.1 summarizes the results of the test. The results are divided into three columns. The first one shows the kind of test: Simple, just one CFDI, and Compound, several

[23] Apache Jena. (2017, Jul. 29). Apache Jena - Home [Online]. Available: http://jena.apache.org.

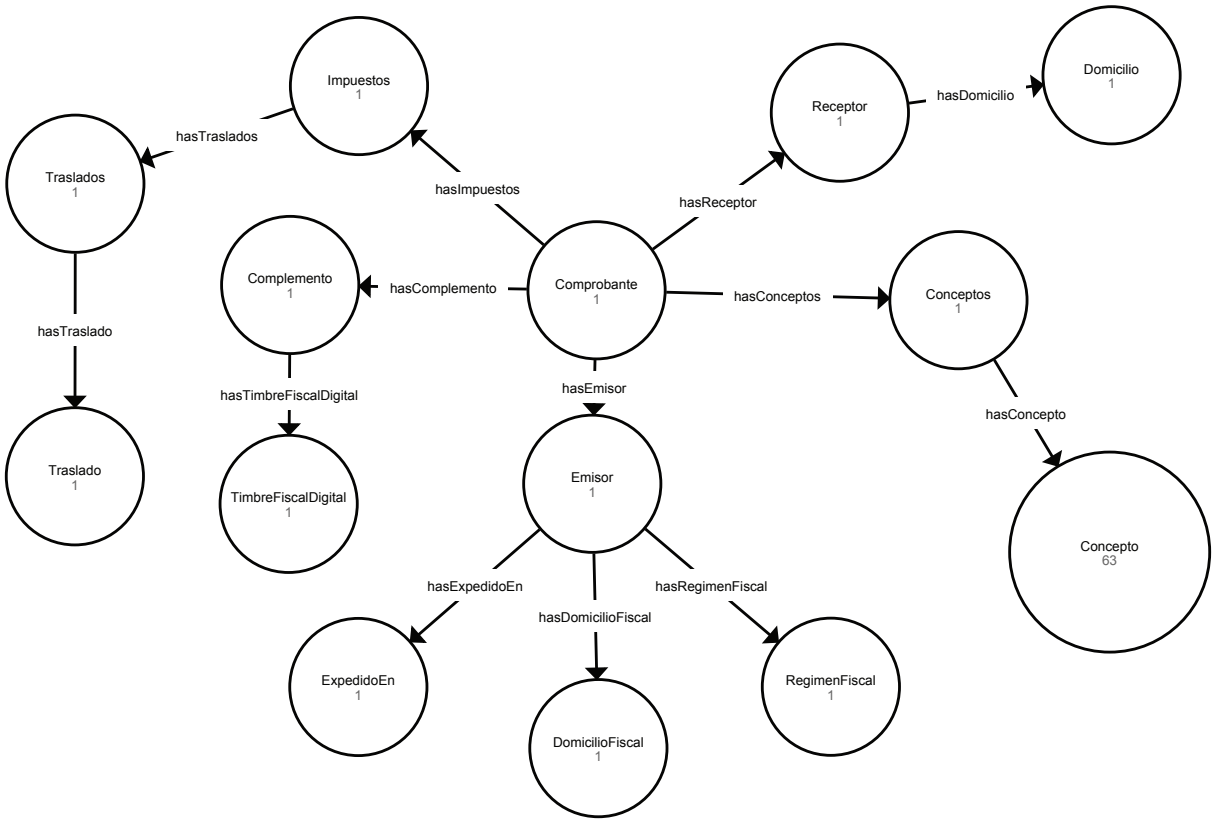


Fig. 2.7 OWL: TBox and ABox relation results.

CFDIs invoices from the same client; in this case, about 10 invoices were used. The second column shows the number of nodes that were detected during the mapping. The minimum quantity of nodes of a CFDI is 30 elements, without considering the instances. More invoices require increasing the number of nodes in the ontology. An invoice can have one or many concept elements, depending on the number of products sold; that is why the number of elements not always increases linearly. Finally, the last column shows the number of triplets constructed during the mapping process. It is interesting to observe in Table 2.1 that the higher the amount of CFDIs, the more duplicity. Once the duplicated nodes are eliminated or linked, a more compact ontology result. Thus, the use and exploitation of the new ontology are simplified. Fig. 2.7 shows the ontology resulting from the transformation process. It can be seen that every concept has a several of individuals linked. It can be considered that the outcome is satisfactory since the ontologies obtained were tested with queries from SPARQL and DL to prove the information remains logical, with low consumption of resources. When a transformation is made and added to an existent ontology, it is necessary to consider the

repetitiveness of concepts, e.g., unique and not duplicated identification IDs. When a transformation is made, and the result is added to an existent ontology, it is necessary to consider the repetitiveness of concepts, distinguishing them between unique and duplicated IDs. Besides, there is no rule to classify the properties in the Data or Object elements, since there is no form to differentiate them.

2.4.4 Future works

By the time this report was finalized, there were some changes in the policies on the Mexican Ministry of Finances that affected the structure of the CFDI here described. The last version of the CFDI[24] (v3.3) has some changes that improve the description of products or services that are included in the sale. Nowadays, they have to be classified based on a universal catalog of products or services, named United Nations Standard Products and Services Code (UNSPSC)[25] code, that holds about 50,000 classifications. This will allow us to include, in future research projects, classifiers of products or services that are capable of detecting relations among invoices and, by doing so, improving the semantic quality of the resultant document. On the other hand, a natural language process strategy is being developed in order to build a semantic QAS so that decision-makers can exploit the knowledge generated and organized in the extracted ontology. This software is currently under development and will be described in a future research report. This work is part of a broader research whose objective is to introduce semantic technologies to small and medium-sized companies in Mexico. Not only the semantic extraction of CFDIs is of interest for this project, but also the use of Spanish language given the characteristics of Mexico's companies, and a simple form to present the information so that users with no or little experience in this technology can exploit the information that is extracted. The ultimate objective is to integrate the resultant research work in an application that can be commercialized in business packages or management tools, taking advantage of the legislative changes that allow people to have access to a significant quantity of information stored in the cloud.

[24] SAT, Factura Electronica, Apr. 2, 2018 http://www.sat.gob.mx/informacion_fiscal/factura_electronica/Paginas/default.aspx.

[25] UNSPSC, United Nations Standard Products and Services Code, Aug. 25, 2017, https://www.unspsc.org.

2.5. Conclusions

A proposal for the automatic transformation of digital invoices to semantic networks was described in this chapter. The transformation implemented from CFDI to ontology was successful, according to the proposed mapping strategy. It was possible to create a robust and compact ontology that allows the usage of semantic components, such as reasoners effortlessly, and efficiently. The developed prototype mapped the CFDI information successfully to a semantic structure in OWL, and the results were verified through SPARQL and DL Queries. We consider the quality of the resulting model high since there was no loss of information. In addition, the prototype had an optimum performance for the hardware and software requirements. The processing of CFDI nodes with descriptions in natural language texts, without structure, is still pending. It is crucial to consider it in future work because it may offer new knowledge that is currently stored only in simple strings of text, without interpretation. This first approach is crucial because it opens a way to use semantic technologies based on structured invoice information, and it will allow in the future to incorporate semantic applications of more complexity, e.g., a QAS could use semantic structures to improve the assertiveness in their answers.

3. Text classification

The following chapter study semantic fields and their connection with the classification of text. The elements and semantic units are discomposed from text to identify features based on their contextual qualities to determine their domains. We also find different forms of classifying a text using statistics, distributional, and semantic methods. We defined semantic evidence and applied it to the text classification to implement a semantic-based classifier to understand the semantic evidence between one word and its corresponding class. We use NLP and distributional semantic (DS) techniques to determine its relationship. We use available ontologies to retrieve semantical traits on the terms to classify according to its classes and reduce ambiguity problems. The primary objective is to take advantage of formal semantic models to enhance classification performance.

3.1. Related work

Since computing syntactic tools emerged from analyzing information, text classification has been evolving and has had several different types of models that were proposed for estimating continuous representations of the words. Bag of words (BOW) [Harris-54], [Sadek-14] and [Bouslimi-13] is a highly efficient model due to its simplicity, economic computing; however, since the order of the words is not kept, there is a significant loss in their meaning or ambiguity.

According to the bayesian classification described in [Friedman-97], [Cohen-16] and [Tang-16], naive bayes classifier is of simple implementation, and it has an excellent computing performance. However, semi-supervised learning requires a large amount of training data to obtain excellent results. The same as BOW, it does not consider syntactic structures and, even less, semantic structures. The vector space models (VSM) [Salton-75] is highly used in practice because they are useful in high dimensional spaces, even with small sizes training samples. They use a subset of training points in the decision function, called support vectors, so they also are memory efficient[1].

Nevertheless, the number of features is much higher than the number of samples and avoiding over-fitting in choosing kernel functions, and regularization terms are crucial. All of them have

[1]Scikit-learn Machine Learning in Python, Jan. 29, 2018, http://scikit-learn.org/stable/.

shown acceptable performance solutions. However, they are based on statistical data to set the relation of a word with the class.

Therefore, the aim of our research work to create a model of classification based on semantic assumptions, started with the statistical approaches to establish a baseline to evaluate future works where the formal semantic techniques are to be used. Although this work does not reach a formal semantic model correctly, it follows its bases as semantic fields and traits.

3.2. Distributional semantics approaches

DS is a branch of study that explores how statistical analysis of large corpora, and in particular word distributions and statistical, can be used to model semantics [Lenci-08]. We have chosen LSA and Word2Vec models because they are two of the most active for word meaning representation, although we do not discard other works such as [Pennington-14], [Joulin-17] and [Levy-14] to mention a few.

LSA [Landauer-97] extracts and stands for the meaning of the words in context through statistics treatments applied to a large body of texts. It was developed to capture hidden word patterns in a text document. It is essential to highlight that the semantic is taken by understanding terms as references. Mapping of discrete entities in a space and a process simplification with a dimensionality reduction are characteristics that make LSA particularly attractive. However, LSA requires a relatively high computational performance and memory in comparison to other information retrieval techniques and the difficulty in determining the optimal number of dimensions to use [Landauer-13].

Word2vec [Mikolov-13a] and [Mikolov-13b], consists of two neural network language models. It uses vector-oriented reasoning between words and defining features of the neural network language model, where the words are depicted as high dimensional real-valued vectors and generate a representation model. It aims to identify contextual patterns that get some semantic evidence. Thus, it develops a framework using a model of recurrent neural networks (RNN) that preserves a track of each analyzed sentence, hence representing the text with remarkable syntactic and semantic properties. However, the model does not have any knowledge of syntax, morphology, or semantics.

Although in a purely statistical manner, in [Altszyler-16], the authors show that semantic relationships are often preserved in vector operations on word vectors. Recent optimizations are

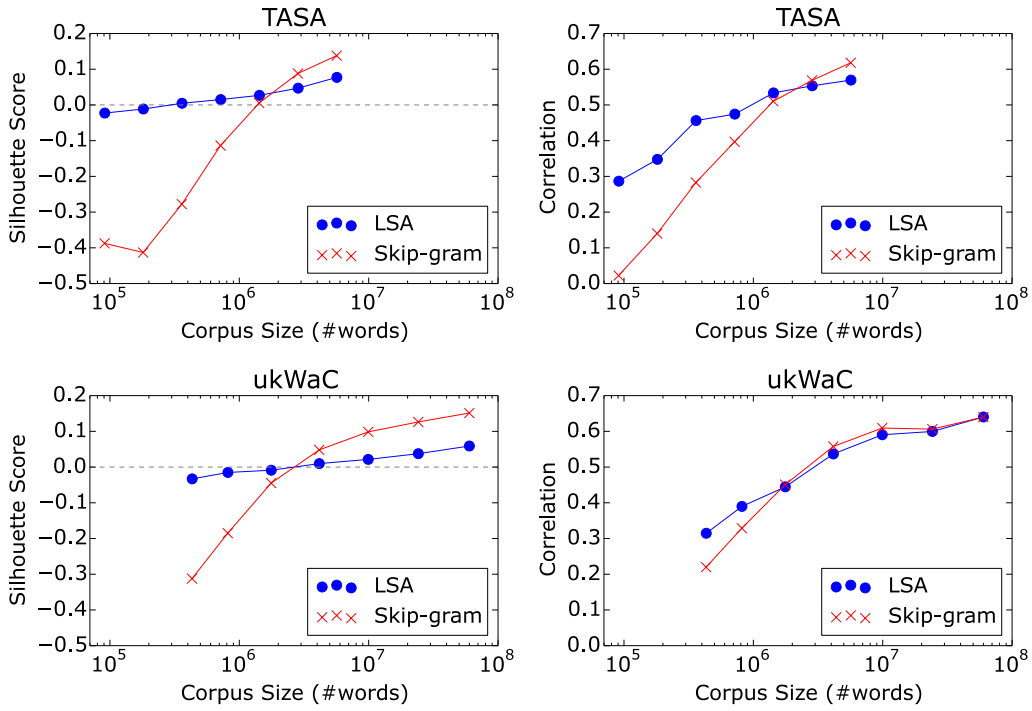


Fig. 3.1 Semantic categorization test performance.

presented in [Kusner-15], where the knowledge encoded in the Word2vec space is integrated with hyper-parameter free, highly interpretable, and naturally incorporated, leading to high retrieval accuracy.

The comparative analysis in [Altszyler-16], [Elekes-10], shows an intrinsic difference between LSA and Word2vec, as illustrated in Fig. 3.1, where left graphs and WordSim353[2] test performance (right graphs) are in function of the corpus size for LSA and Skip-gram model, the size of the different corpus is considered as the number of tokens that they contain.

Word2vec is used in our proposal since it obtained better results in prediction inference. However, it should be mentioned that Word2vec's performance decreases with small datasets, and it needs several training data to fit its high number of parameters.

[2]WordSim353, Similarity and Relatedness, Dec. 15, 2020, http://alfonseca.org/eng/research/wordsim353.html.

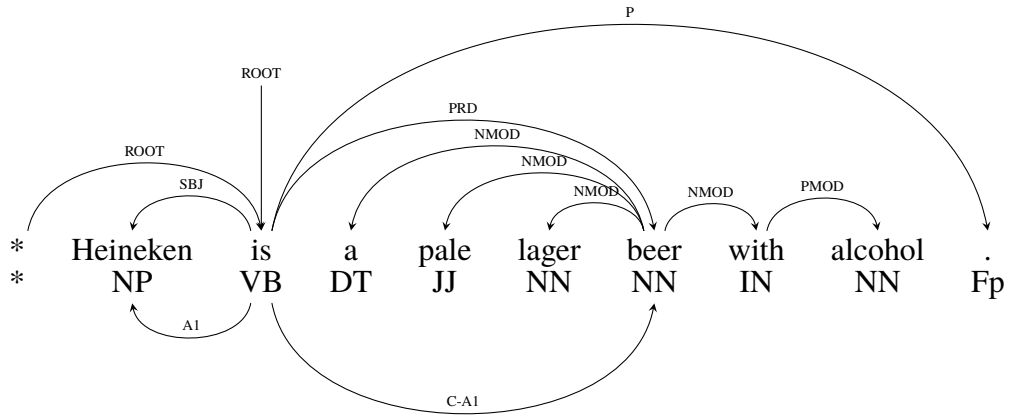


Fig. 3.2 Dependency parser: sentence analysis.

3.3. Grammar approaches

Although the NLP tools are not text classifiers, they are an indispensable tool for the classification task. Here, the POS, named entity recognition (NER), preposition–sense disambiguation, and dependency parsing tasks are used to detect intrinsic cues based on the word itself and extrinsic cues based on its context.

3.3.1 Named entity-recognition

The NER task is beneficial for the extraction and recognition of entities, where finding elements such as names of people, companies, locations, or even numbers are critical parts within the process of generating semantic proxies. The recognition techniques of entities in their beginnings were implemented by using handcrafted ruled-based algorithms, but modern approaches use the most common machine learning techniques. Since the NER frameworks are easy-to-use and efficient, nowadays are part of more complex language analysis. The early implementations had their beginnings by using models of *information extraction systems*, such as Gibbs sampling [Finkel-05]. However, recent works found that *maximum entropy* could have better results [Chieu-02].

Nevertheless, there are several challenges, due to the language complexity and the diversity of linguistic phenomena, especially on the tasks of recognition of entities like Metonymy [Levrat-16]. Nevertheless, the biggest problem is that most of the tasks of recognition of entities put aside the

context of the text that is being analyzed. Works like [Williams-17] about NER in *Challenging Context* will be of great importance. In the text: "The automotive company created by [Henry Ford] in [1903]," Henry Ford is detected as an entity of a person's type name, 1903 as a year; but, entities like Ford Motor Company are unlikely to be found.

3.3.2 The preposition-sense disambiguation

The preposition–sense disambiguation is potentially useful in the extraction of semantic proxies classifying instances of polysemous words into their proper sense classes. Despite its great importance, the disambiguation between preposition senses has not been an object of considerable study. Some works, such as [Tratz-09], suggest that the maximum entropy classifiers have been well suited for disambiguation tasks, and so, motivated features can improve the accuracy of preposition sense disambiguation.

3.3.3 Dependency tree

In this model, the predicate of a sentence can be obtained from an analysis of the dependencies attached to a sequence of words that are related to it, particularly the subjects. In this way, it might be possible to recognize relationships in binary words and create a semantic graph where triplets are detected; but to achieve this, it is not enough to have triplets. The ontologies require to be modeled using semantic rules of the original text. This is why additional support methods are required to identify taxonomies and named entities, as seen in the previous section. The work in [Goldberg-17] presents a series of improvements in the classifiers using NLP techniques. They use a classification based on parser of grammatical dependencies, dependency tree, or arc-factored parsing; Fig. 3.2 shows an example of this parsing. There are good results that come out from this last work. However, as its author mentions, the dependencies analysis is not always accurate. They are subject to change along the way the statement is described, so it is necessary to add some semantic techniques to obtain better results.

3.4. Semantic approaches

This work regards the semantic approaches to the ones that are based on the explicit relation of their elements. Usually, their structures are formed by graphs or ontologies. These models are of high interest since they are based on pre-established hierarchies and rules that favor the results of the text classification. There are a few text classification studies that use isolated semantic analysis to classify. However, approaches such as [Liu-98] compared with the probabilistic statistical model, the association rule method pays more attention to utilize association relation among features for classification. Other approaches use WordNet [Miller-07], a lexical–semantic network in which nodes correspond to word senses. Séaghdha [Seaghdha-09] uses graph–based kernels [Shawe-Taylor-04] on WordNet for classification and attains excellent performance according to SemEval-2012[3]. Up to now, the previously described approaches achieve excellent results in specific domains. However, they are limited because they use pre-built structures with rules to succeed, and supervision is required. Compared with them, this study approach generates semantic structures of texts in natural language that serve as features in the text classifications and avoids limitations of the approaches previously described. We combine the previous DS approaches and the described NLP tasks to achieve better results, together with the ontologies and the association-rules methods, to have a new classification model. The following section describes a fundamental concept for the generation of semantic structures or proxies.

3.5. Semantic evidence

The semantic evidence is linked to the existence of enough connections between the features of two or more elements for sustained evidence and a semantic related to their meaning, the meaning of the elements compared. Such connections are obtained by measuring the space and distance between the elements. Tools of semantic measure estimate the strength of the semantic relationship between units of language through a number according to the comparison of information supporting their meaning. In other words, semantic evidence is depicted by a value that indicates the level of relationship between one text and its features. Such a comparison can only be measured with

[3] SemEval-2017 Task 2. Multilingual and Cross-lingual Semantic Word Similarity, Jan. 5, 2018, http://alt.qcri.org/semeval2017/task2/index.php?id=task-details.

semantic evidence, as will be shown later.

The semantic measures compare the relatedness or the similarity of one, or more than one, of the elements. Harispe et al. [Harispe-15] define the semantic relatedness as the strength of the semantic interactions between two elements with no restrictions on the types of the semantic links, as well as the semantic similarity as a subset of the notion of semantic relatedness only considering taxonomic relationships in the evaluation of the semantic interaction between two elements. Thus, to extract semantic evidence, there is a great variety of measuring metrics that may be used. However, all of them, as a whole, are based on strengths and relations.

Once the evidence is obtained, semantic proxies can be generated, which are the features of a word, originated by the existence of the relations between the semantic elements, e.g., words and concepts; that is, what similar properties exist between the terms based on their meanings. It is essential to mention that the quality of the proxies depends, on a high degree, on the techniques that these semantic measures use.

3.5.1 Similarity measure

In this chapter, semantic evidence that is sustained on the similarity of a word is used. The measures between words and features are obtained through semantic measures models that are the approaches designed for comparing semantic entities, such as units of language, e.g., words, sentences, or concepts and instances defined into knowledge bases. Tversky, in his *studies of similarity* [Tversky-78], proposed a *feature model* that can be used to analyze the similarity relations between words according to a feature–matching function F, which makes use of their common and distinct features. Thus, the similarity is formally described as

$$sim_F(u, v) = F(U \cap V, U \setminus V, V \setminus U) \tag{3–1}$$

where U and V are sets of features. F increases when common distinct features are added or removed. Therefore, for the aim of the present study, only the featured shared are used. The differences between U and V are excluded since, in the vector, they represent distant elements. There is a lack of meaning in common, also known as semantic scarcity or non-semantic elements. Tversky [Tversky-78] proposed the ratio model $sim_{rm}(u, v)$ to compare two objects U and V represented through sets of

features U and V,

$$sim_{RM}(u,v) = \frac{\alpha f(u \cap v)}{\alpha f(u \cap v) + \beta f(u - v) + \gamma f(v - u)} \tag{3–2}$$

The symmetry of the measures produced by the two models can be tuned according to the parameters α and β. This enables the design of asymmetric measures. The primary constructs of the feature model are the function f, which is used to capture the salience (an outstanding feature of a stimulus such as intensity or frequency) of a feature or set of features [Tversky-78]. Therefore, the operators, $\cap$ and $\cup$ are based on feature matching F, and the function f evaluates the contribution of the common or distinct features to estimate the similarity. In this approach, the ratio model similarity is used as a possible way of normalization. Figuring out the similarity according to *ratio feature model*, it happens that if $w_1 = beer$ and $w_2 = soda$ is compared (their features are extracted from Wikipedia sources), then it is obtained that,

$$A = \begin{bmatrix} barley \\ yeast \\ alcohol \\ hop \\ fermented \end{bmatrix}, B = \begin{bmatrix} sugar \\ carbonate \\ drink \\ sweetened \\ flavorings \end{bmatrix}, A \cup B = C \tag{3–3}$$

a matrix that includes features of each term is generated, and each of the term-feature distances is determined by

$$D = \begin{bmatrix} c_1 & p(w_1,c_1) & p(w_2,c_1) & \dots & p(w_n,c_1) \\ c_2 & p(w_1,c_2) & p(w_2,c_2) & \dots & p(w_n,c_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_n & p(w_1,c_n) & p(w_2,c_n) & \dots & p(w_n,c_n) \end{bmatrix} \tag{3–4}$$

so, it has the following result

$$D = \begin{array}{c} \\ c_n \\ \left[\begin{array}{c} \textit{barley} \\ \textit{yeast} \\ \textit{alcohol} \\ \textit{hop} \\ \textit{fermented} \\ \textbf{\textit{sugar}} \\ \textit{carbonate} \\ \textit{drink} \\ \textit{sweetened} \\ \textit{flavorings} \end{array}\right. \end{array} \begin{array}{cc} \textit{Beer} & \textit{Soda} \\ p(w_1, c_n) & p(w_2, c_n) \\ \hline 0.529 & \mathbf{0.346} \\ 0.517 & \mathbf{0.386} \\ 0.555 & \mathbf{0.435} \\ 0.512 & \mathbf{0.365} \\ 0.600 & \mathbf{0.493} \\ \mathbf{0.527} & \mathbf{0.444} \\ 0.555 & 0.580 \\ 0.771 & 0.562 \\ 0.528 & 0.487 \\ \mathbf{0.443} & 0.451 \end{array} \left.\begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{array}\right] \tag{3–5}$$

the characteristics form the first column; the second column is the result of the similarity between word 1 and characteristic 1; the third column is the similarity between word 2 and characteristic 1.

Then, just the values with an acceptable similarity degree must be kept, where, the longest distance between the features will be fixed as an acceptable limit, e. g., in the matrix D, the closest relation distance that exists between sugar respecting soda is 0.444. Any lower value will not be considered, in other words, there is a lack of semantic similarity. At last, the array is sorted according to the model of features; to do so, the effective terms features are turned on in order to obtain the following binary arrays:

$$u = \{1, 1, 1, 1, 1, 1, 1, 1, 1, 0\}, \tag{3–6}$$

$$v = \{0, 0, 0, 0, 1, 0, 1, 1, 1, 1\} \tag{3–7}$$

where $\alpha = 2$, $\beta = 1$ and $\gamma = 1$, and f is the cardinality of sets,

$$u \cap v \textit{ is the set of positions } \{i|u_i = v_i = 1\} \tag{3–8}$$

$$u \setminus v \textit{ is the set of positions } \{i|u_i = 1, v_i = 0\} \tag{3–9}$$

$$v \setminus u \textit{ is the set of positions } \{i|u_i = 0, v_i = 1\} \tag{3–10}$$

using the ratio model formula, we found that

$$sim_{RM}(P, Q) = \frac{2|\{5, 7, 8, 9\}|}{2|\{5, 7, 8, 9\}| + |\{1, 2, 3, 4, 6\}| + |\{9\}|} = \frac{8}{8 + 5 + 1} = 0.615 \tag{3–11}$$

when $\alpha = 0$ instead of a similarity function, the result is a distance function. Table 3.1 shows a

TABLE 3.1. AVERAGE SIMILARITIES FOR PAIRS OF WORDS.

p	q	$sim_{RM}(p, q)$	$sim_{L_P}(p, q)$	$sim_{COS}(p, q)$
Juice	Drug	**0.722**	0.690	0.820
Television	Tires	**0.400**	0.223	0.320
Computer	Refrigerator	**0.330**	0.023	0.321

comparison between similar pairs of words. Thus, the result is a similarity index (SI) based on features. For this example, features extracted from a vector of Wikipedia are used; however, it is possible to use n-features from different sources. It is necessary to consider that the semantic evidence factor (remoteness-indexes) must be the same (homogenized) in all the cases.

SI is essential to filter features. It varies between 0 and 1, where the closest to 1 is the most similar. One hypothesis is that if two synonyms are taken, SI should be equal to one. although it would still be subject to discussion. Despite of these limitations, SI is well understood, and it works well for all practical purposes since it simplifies the ontologies construction.

There are more measures similar to this model. The most outstanding are Minkowski sim_{L_p} distance metric [Ding-11], cosine similarity sim_{COS} [Sidorv-14], and Pearson's product–moment correlation sim_{PEA} [Shevlyakov-16]. However, in our study, there is no much improvement if a more complex variation of the model is used. The model proposal of this work uses the semantic

evidence model as binary units to determine the meaning, understanding this meaning as the distance and closeness of a word concerning another one.

These measure metrics do not consider the context nor logic questions as inferences, though, perhaps there might be some more complex models where these aspects are considered. The models considered are based on simple mathematical approaches. However, they can be used as a starting point for more sophisticated analysis, such as the ones of language processing.

In the previous sections, we reviewed several techniques to classify text. In our experimental study, we found that support vector machines yield the best results in quantitative scenarios. Results are omitted for the sake of brevity. With vector machines, the distances found between words for the relation in a determined text might create the empiric effect of semantic evidence required to have a similarity measure. However, although it has good results, this effect lacks a semantic analysis of the terms measured. Critical linguistic phenomena, such a context, must not be left aside.

3.5.2 Context and other challenges

From the closeness between words in a text, it is possible to obtain semantic features. These features sometimes are not so evident even when using techniques of similarity between words, which means that as the vocabulary grows in size, there will also be more relations between words. Therefore, this vocabulary will have more semantic richness. Based on the statement of Lewis [Lewis-70], it will be assumed that the semantic is meaningless if there is not any context; that is, there is not any semantic evidence. In this case, a word is transformed into a tautology, which means that if there is not any context, every statement is true. It is impossible to validate the semantic evidence.

The context is affected by the distance between two words when the base vector is changed. Fig. 3.3 shows two graphs: (a) corresponds to WordNet and (b) to Wikipedia. It should be mentioned that both vectors were created with different corpora and algorithms. It can be observed that WordNet has more accuracy than Wikipedia due to the nature of both ontologies: WordNet is a concept ontology, while Wikipedia contains both concepts and instances in its corpus. This method aims to test the classification methods previously reviewed [Escobar-18]. We elaborate a model and compare its results with commercial, open, and closed domain classifiers. We use SemEval to measure the degree of semantics evidence extracted. Thus, we can have the best comparison point, estimate the

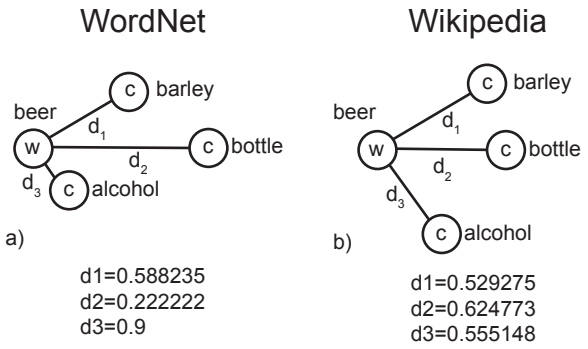


Fig. 3.3 CFDI document: electronic invoice with levels of nesting.

percentage of effectiveness, and determine if the methods succeeded in a similar way as the one obtained through a formal, non-automatic, semantic analysis.

Regarding the preparation of the components, all the vectors were created by using the word mover's distance (WMD) algorithm [Kusner-15]. The corpora came from different sources, mainly Spanish corpuses such as Ancora lexicons [Taule-08], SenSem (Sentence Semantics) , Real Academia Española (RAE) , and Wikipedia. Besides, Wordnet version 3 was used as a similarity framework and as a validation method in the WMD vectors; however, WordNet is not finished in Spanish, so a translation English–Spanish was required to enable the language. The algorithms of fuzzy lookup, label lookup [Zhang-14], [Spitkovsky-12], and [Bast-15] for the entity matching were used to cleanse the terms. For the natural language processing, the Stanford CoreNLP framework was used, it was beneficial that the 3.9 version included the Spanish universal dependencies.

3.6. Semantic classification model

The semantic classification process is broken into three phases; Fig.3.4 shows the process flow: (1) features extraction, (2) semantic evidence, (3) comparison to the pre–classified catalog. In the first phase, we use encyclopedic reference sources to verify that the word we will classify exists in the Spanish language. There are two types of sources: formals, which means they belong to an organization or community that formalizes the word. It assigns one or more definitions, for example, dictionaries or encyclopedias. Once the word is found in this kind of repositories, it is considered as a conceptual element. Otherwise, it may happen that the word is not a concept; then, an open-source of reference is used, that is, of the public domain such as Wikipedia, Freebase, or the like. Moreover,

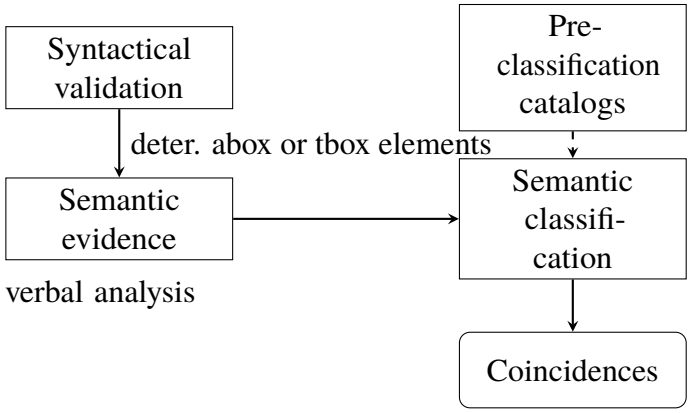


Fig. 3.4 Semantic classification process.

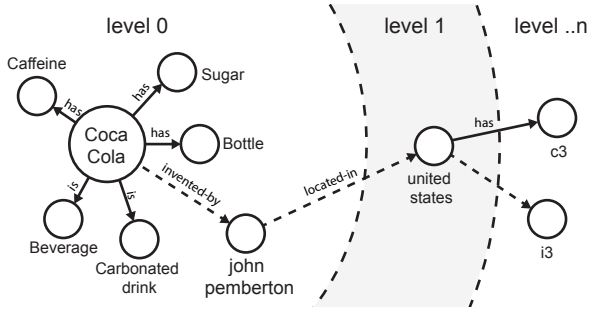


Fig. 3.5 Result of the extraction of features.

if the element is found, it is considered an instance, also known as an individual. The second phase takes information out of the definitions of the words to be found, searches for semantic evidence in the terms, and generates semantic proxies. The semantic evidence is extracted from the open sources of reference such as Wikipedia or any other such as Google, Freebase, or Yago[4], to mention a few. The reference text is cleaned. By using NLP, it is possible to find the most important entities of the text. Nevertheless, not all entities will be found. Up to now, the corpus of the named entity recognition (NER) has not been trained for the text. However, if the term that is being searched is not found in this task, the percentage of obtaining entities with quality is lowered. So, by using POS tagging, the nouns are extracted. By themselves, the nouns could be independent of the first term, one that is being searched. To find its dependencies, a search in the levels of given dependencies analysis is required. However, it was learned that the deeper in the scale, the higher is the exponential growth of the computing process. For this model, only one level of relation was used, e.g., in the text "Coca-Cola is a sweetened beverage elaborated by The Coca-Cola Company Inc." only the predicate

[4]Yago Project, Dec. 15, 2020, https://yago-knowledge.org.

"beverage" is taken, while "elaborated by The Coca-Cola Company Inc." is not considered. The result is a semantic graph-based structure of the term regarding its definition, as illustrated in Fig. 3.5, the semantic graph whose main *class/instance/node* is the pre-classified term. The graph may be formed by to more classes or instances.

The third phase relates and compares the second–phase created graph with previously classified data. If there are significant coincidences, it is considered that there is semantic evidence, and it is classified with the name of the pre–made, compared node element. Up to this point, there is a graph and a matrix. It is just a matter of choosing the values of higher rank and assign them a label classification. We can use linear classifications that do not require external information. There are several methods of classification; however, after trying *decision trees* and *naive Bayes* [Lewis-98], it was determined that the classification for MaxEnt [Nigam-99] has better performance for this type of text classification, mainly because it admits distribution of the Bernoulli form. The MaxEnt formula is described as:

$$p(c|d) = \frac{1}{Z(d)} exp(\sum_i \lambda_i f_i(d,c)), \tag{3–12}$$

$$Z(d) = \sum_c exp(\sum_i \lambda_i f_i(d,c)) \tag{3–13}$$

where each $f_i(d,c)$ is a feature, λ_i is a parameter to be estimated, $Z(d)$ merely the normalizing factor ensuring a proper probability, $p(c|d)$ denotes a set of all conditional probability distributions, and finally, d and c are documents and characteristics, respectively. A set of weights are parameterized which combine the *joint-features* that are generated from a feature set by an encoding. The encoding maps each (feature-set, label) pair to a vector. Here, to compute the probability of each tag following (3–12) we use:

$$prob(p|l) = \frac{dp(w,\ \textit{encode}(p,l))}{sum(z(w,\ \textit{encode}(p,l))\ \textit{for l in labels})} \tag{3–14}$$

where l = *labels*, w = *weights*, p = *features*, and z, from (3–14), is the dot product given by:

$$z(ab) = sum(x * y\ \textit{for}(x,y)\textit{in zip}(a,b)) \tag{3–15}$$

here zip returns an iterator of tuples. For example, if *Heineken* is to be classified based on the catalog

of products of the UNSPSC[5], it is necessary to extract from Wikipedia a page definition, where the information retrieval task returned two proxies,

$$labels = \{barley, beer, alcohol, bottle, pilsner\} \tag{3–16}$$

and, from the UNSPSC catalog, five potential classes are found:

$$UNSPSC = \{beverage, substance, music, toys, drugs\} \tag{3–17}$$

Now, the similarity M that was obtained from each classifying-label is

$$M = \begin{array}{c} \begin{array}{ccccc} label & & p_1 & p_2 & p_3 \end{array} \\ \left(\begin{array}{ccccc} beverage & & a:0.549 & b:0.407 & c:0.301 \\ substance & & a:0.310 & b:0.298 & c:0.335 \\ music & & a:0.211 & b:0.229 & d:0.263 \\ toys & & a:0.291 & b:0.244 & e:0.355 \\ drugs & & a:0.271 & g:0.239 & h:0.208 \end{array} \right) \end{array} \tag{3–18}$$

Notice that features p may vary according to each classifier; thus, the beverage contains the labels a, b and d while the classifier substance contains the label c that is unique. Classifiers with less similarity, such as drugs, have features g and h. The amount and heterogeneity of classifiers do not affect the model of MaxEnt and allows us to use classifiers yet with different labels. Finally, a MaxEnt is applied to the M matrix, and the result is:

$$\begin{array}{c} \begin{array}{cc} label & score \end{array} \\ \left(\begin{array}{cc} beverage & \mathbf{0.77} \\ substance & 0.43 \\ music & 0.27 \\ toys & 0.25 \end{array} \right) \end{array} \tag{3–19}$$

If all the features were turned on, the result would be a beverage with 1.000. That means that what

[5]UNSPSC, United Nations Standard Products and Services Code. Aug. 25, 2017, https://www.unspsc.org.

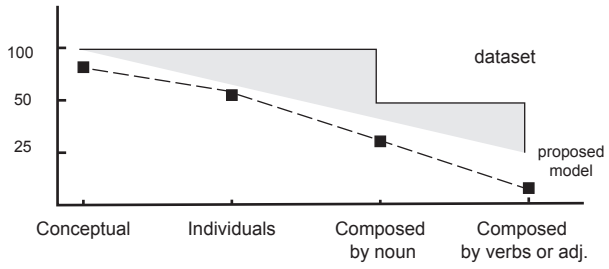


Fig. 3.6 Results of the semantic classification.

is intended to be classified is the same to what is classified. However, in this case, it matches only 77%; that is, *Heineken* is 80% similar to a "*beer*". Here, it is possible to see that the label "*beverage*" is the one that has a better score regarding the other options; Therefore it is chosen as the classifier.

Lastly, in case of having groups and family-level of classifiers, MaxEnt can be used in a series of iterations, where the pre-configured catalogs are organized in groups. That means that every family-level requires a sub-series of new classifications using (4-14).

3.6.1 Results

To test this model, we built a prototype that classifies products and services according to the UNSPSC. The aim was to know how the model performs in a real environment where the words to be classified can get connected with some previously defined catalogs. The words to be classified were uploaded by users that ignore the way that the model works, and they require to know a product or service identification (ID) code to be used later in some other processes.

In this context, a data set with 100 terms to be classified is built. These are divided into four different types: (1) conceptual, e.g., *beer, ball, car or factory*. (2) Individuals, such as beer brands, e,g., *"Heineken," "Budweiser," or "Corona."* (3) The terms composed by nouns, e.g., *"AeroCanada" or "Economic politics"*; and finally, (4) the most complex group is formed by composed verbs, e.g., *"computing in action," "eating fast,"* or adjectives, like *"European Economic Community," "bolt nut," "pipe wrench,"* among others.

UNSPSC is used as a pre–catalog in which the terms should be classified. However, due to its extension, about 50,000 terms, it is not possible to categorize all the words. The categorization is possible with 1000 words clustered in seven families. Fig. 3.6 shows that the more complex is the

```
SELECT ?subclasseLabel
WHERE {
  wd:Q854383 wdt:P31 ?entity.
  ?entity wdt:P279|wdt:P31
  ?subclasse.
}
```

Fig. 3.7 SPARQL query from Wikipedia page.

term, the less successful could be the classification due to a higher effort required to process the text.

For the semantic evidence, different sources are used, such as Wikipedia, WikiDic, RAE, and WordNet. A classifier works well when the ambiguity of the term is low; that is, words like concepts, places, organizations, and proper names are classified with a high percentage of closeness. In other cases, with higher ambiguity, like product brands and company names, the accuracy of the classification is lower, and sometimes the word could not be classified under only one classifier. There is a issue of processing words or terms that have a large number of semantic characteristics, over 1000; in these cases, the time elapsed in getting the results is notorious because of the number of terms to be found. To diminish execution time, the MaxEnt algorithm classifier [Gzyl-95] is pre-charged in memory.

When the word is not a concept, it is necessary to make a pre-analysis to identify the type of concept or instance where it belongs. In this case, the word is known as an individual. The determination of an individual must also pass through the process of contextualization, identifying the method of extraction of characteristics and the analysis of contextualization. Both of them are part of the pre-classification.

3.6.2 Underlying issues

Throughout the implementation of this model, some issues were detected. Although some measures were taken to prevent them, they are mentioned here to consider them for future work.

A. Enhancing proxies

Before determining any semantic method, it is necessary to create high-quality proxies of a

text where the definition of the term has been found. Such sources can be structured or raw. For structured sources, Wikipedia generates *InfoBoxes* that contain classified information of the article that is being read. However, it is necessary to analyze which attributes of the *InfoBox* are good and which are going to be discarded. In Fig. 3.7 a query in Wikidata is also used to execute a SPARQL to provide structured information from the Wikipedia page. What is being searched is mainly instances, where encences $= x|x \in e$ and $x \in t$, being e an entity and t a relation or a sub-class. Authors in [Lehmann-15] propose useful approaches to consider in future work. When it is a raw source, the problem is to find what part of the text holds the description of the term that is wanted. In our work, heuristics and paraphrasing are used to detect the explanation of the searched word. Nevertheless, this depends, to a high degree, upon the author's narrative.

B. High number of categories

By increasing the number of categories and classes, the performance of the system decreased. It is necessary to partition, fragment, or index the types to decrease this problem. This same phenomenon is observed when the extraction of semantic evidence for a term produces a high number of features.

C. The bundle theory

This model may fall in the same problems as in the *bundle theory* [Rastier-96], where the result of classifying based on a set of referential features may vary according to the information sources. It is not known if the extracted elements are determinant, neither is known whether all the items must be present for the object to be identified. If it is required to elude these difficulties, it is necessary to rely on truth trials on the elements of the bundle. That is, to know whether this or that element belongs or not to this or that object. To distinguish systematically among content, concept, and referent is a must. Summarizing the above, it is possible to classify words by using semantic measures. It is possible to use different resources and compare the results and then to have a better categorization and, as a

consequence, a better classification. However, the classification remains limited, and it works well with isolated words. It is pending for future works to make some adaptations to improve the process and, therefore, to identify compound words and enhance the scope of the classification. We can improve the retrieval process by combining formal semantic techniques such as compositional and interpretative because this technique gives a more understanding of the language meaning than the syntactic and statistical approaches.

3.7. Algorithms

This section describes an implementation of the classifier previous defined, regulated by the standard UNSPSC from the united nations (UN).

3.7.1 Core algorithms

Our first step in this algorithm is to validate the word. Validation aim at determining if the term is an existing word, if it is spelled correctly, if not, we used edit-distance [Levenshtein-66], lemma and stemming [Porter-80] as correction techniques, as well as determine if the term is a concept or an instance. Wikipedia APIs can be consulted in order to obtain spelling suggestions for a concept, that is, misspellings. The vectors are previously generated using the algorithm [Mikolov-13a] and [Mikolov-13b] that allows operations of vector inference. It includes stopping words, misspellings, commonly paired words, that in the future have to be removed from the corpora. Real academia española (RAE) was taken as a conceptual reference, even though any other encyclopedia such as dict.com, Oxford, or the Wordnet dictionary could be suitable options. RAE is a complete corpus for the Spanish language, holding over 300,000 terms. The UNSPSC provides a logical framework for classifying goods and services. The last version of the UNSPSC catalog is pervasive, 65,000 categories; however, a 1000–element sample was taken. The result of this classification is a graph with the classifier and the features of every single element. The tuning tasks consisted of filtering the found terms where there is few or null semantic evidence. A consequence of using VSMs is that all the words that exist in the vector are related among them, and therefore, they have a score. An index was defined to determine when a word was so distant that it loses all semantic value for the task in hand; a value of 0.2 indicates the nearest distant possible. The main component of the application is

the classifier. Mikolov's formula [Mikolov-13b] is normalized when the score is figured out

$$score(w_i, w_j) = \frac{count(w_i, w_j)}{count(w_i) \times count(w_j)} \tag{3–20}$$

So, the values of each score are taken and are being added to each label of the classifier. Adding may not be the best option, but just now, it was not possible to find a better index, and it is computationally economical and straightforward.

Algorithm 1 Calculate similarity (x, y).

Input: r_i, s_j *// r_i and s_i are two inputs of array strings*
Output: $classifier.label$, $sem.measure$

// s_j are classes
while $c \leq s_j$
  // r_j are labels
  while $l \leq r_j$
    // exclude words that are not in the word2vec
    if x is where $\{x \mid x$ in $vocab\}$
      // just values higher than .2
      $f \leftarrow$ **SIM**(x, y) *// as a float*
      // this method measures the similarity between a pair words, formula 1
      $f \leftarrow f + f$ *// increment the semantic similarity value*
      $t_j \leftarrow f$ *// add f to t array, labels are missing!*
    end if
  end while
  $train_j \leftarrow t_j + s_j$
end while
// now we have a matrix to train the MaxEnt model
$classifier \leftarrow$ **MEC**$(train_j)$ =0

Algorithm 2 Generate arrangement for the classification task.

Input: r_i, s_j *// r_i and s_i are two inputs of array strings*
Output: $classifier.label$, $sem.measure$ *// as list*

while $c \leq s_j$
  $f \leftarrow$ **SIMCLASS**(x, y) *// as a float*
  // remove x from set s_j
  $r_j \leftarrow f$
end while
// now we have a matrix to train the MaxEnt model
$classifier \leftarrow (train_j)$ =0

Algorithm (1), shows the function to compute similarity. Here, two arrays are taken, R that corresponds to the labels, and S that refers to the classes. It iterates, over classes, and the measures related to the labels are obtained with function *SIM*. This function receives two arrays of feature-words and returns a similarity index. The filter factor has been tuned in 0.2 to avoid that the distanced words be considered, and so, the calculation time is lowered. In line 10 of algorithm (1), we can notice how the similarity score is incremented by adding all the labels for one classifier. In this way, the training set T is obtained. It is essential to mention that in every iteration of classification, a new training set is generated. Finally, *MEC*, this function receives a training set array and returns a sorted arrangement of classes, is executed to obtain the label of the best classifier, based on the best distribution of biased probability, that means, the one that maximizes the entropy.

The second algorithm of classification is shown in algorithm (2). It completes the functionality of algorithm one. It generates an arrangement of the classifiers sorted from higher to lower according to the distance of the terms or the semantic measure.

TABLE 3.2. UNSPSC LEVELS AND DISTRIBUTION.

Level	Quantity	Success	Complexity
Group	100	80%	low
Segment	100	68%	low
Family by nouns	50	38%	low
Class	25	8%	high
Product	25	8%	high

3.8. Evaluation

In order to evaluate this classifier, a data set with four different text groups, 100 concepts, 100 individuals, 50 composed terms were created. Every single concept was manually classified according to the UN catalogue, and divided into five different groups. Table 3.2 shows the distribution of terms according to the UN standard. The UN catalog of classification has about 50,000 elements classified in three levels in which every element has an assigned ID. This catalog is used in Mexico

for generating invoices or electronic receipts to back up a sell or purchase of a service or a product.

The execution of the experiments was done by using only one terminal, where all the vectors were taken as web services. To support the load, it was necessary to use a computer with 24 GB of RAM memory, 120 GB of hard disk with an Intel core i5 processor, at 3.4 GHz. Since the vectors were previously processed, the processing factor was not relevant to the trials. However, other factors hindered the execution of the trials, e.g., The APIs of Wikipedia and Wikidata are limited to a certain amount of transactions per minute. To reduce this issue, a downloading cache was added, so that the problem was reduced. Nevertheless, it is recommended that in future works the corpus of Wikipedia and Wikidata be downloaded in order to reduce the latency of the web.

3.9. Results

TABLE 3.3. RESULTS OF THE CLASSIFIER OF WORDS.

Type of term	Quantity	Success	Complexity
Conceptual	100	80%	low
Individuals	100	68%	low
Composed by nouns	50	38%	low
Composed by verbs and adjectives	25	8%	high

Appendix C, shows the first 20 terms that were used in the test task. Table 3.3, shows the results of the first classification. As seen, the classification of concepts is the one that has the best results. The classification of individuals goes as low as 80%; however, there are some critical ambiguity issues in here, e.g., the terms "*Heineken*" could be understood as a company, a product, the last name, or as a city. Due to this, the classifier is limited to products and services, and heuristics were included to just filter these two groups. The classification of compound concepts, that is, concepts that contain two or more nouns, or those that are formed out of verbs, are much more difficult to be classified.

It was found that the task with the most significant complexity is the extraction of semantic

features. The space vector models do not consider the language issues; that means, they do not interpret the meaning of the words. The extraction of features with purely statistical methods have a good result even without having a semantic understanding. However, it leaves out sophisticated features, as described above.

3.10. Future work

In future work, we will address context and other challenges in more detail. We can also state that vector machines, along with NLP processes, are essential to find more semantic evidence and generate proxies with more semantic meaning. Additionally, techniques of formal semantic analysis could improve the tasks of generating proxies. In future work, we will also develop a model of classification based on the semantic evidence that was obtained up to this point.

3.11. Conclusions

The proposed method reached a performance of 80% over the data to be classified. However, as more complex data is added, such as compound terms, its efficiency drops drastically. Although we found that classification and vectors work well, we can improve the efficiency of the classifier in general by applying formal semantic methods and improving the understanding of the meaning of the term or word. This finding enhances the extraction tasks, generating proxies with semantic characteristics related to labels of a term. So, we will have labels more concise, with higher scores in the similarity of the vector models.

4. Interpretive semantics

Text classifiers extract the characteristics of the words to be classified using statistical methods. Statistical approaches are essential for the search and extraction of information. However, its use is limited to the quantitative symbolism of the words and leaves out the meaning, missing many vital clues in the classification. In this chapter, we use interpretive semantics to improve purely quantitative approaches. To experiment, we developed a text classifier that uses an interpretive model to obtain the elements of meaning and their relationships. Our goal is to find an interpretive position between one text and another previously classified with statistic approaches. We also formalize essential interpretive elements and explore semic analysis. The result of our approach is a semantic feature extractor for classifying text. We conclude that interpretive analysis, far from replacing statistical analysis, can be combined by significantly improving the results of text classifiers.

4.1. Introduction

In the text classification process, the extraction of word characteristics is limited when only language manipulation processes are used. Still, Natural language processing incorporates semantic tasks or processes that use identity models[1] to find semantic units. This implies that the context is unknown in recognizing of semantic units. The semantic unit must be constructed and structured based on the meaning necessary for the context, increasing the semantic element's formation.

There are many ways, methods, and models in formal semantics to analyze the text's meaning. We have chosen interpretive semantics because it is the only theory that focuses on understanding the text as a unit and its relationships, where the reader works or operates. Also, it fits tight with the structure of the ontology.

Interpretive semantics does not display information about meaning but instead explains the meaning of the text and determines the interpretation within a context. Something fundamental about this theory is that it avoids the characterization of the coding processes. The selection of the sense of interpretation is more flexible and ambitious than other interpretive theories and it can

[1] Semantic identity save primary identity data about a resource through a combination of the data with each other.

focus on the subjectivity of the linguistic context [Rastier-96].

Interpretation is a theoretical and methodological paradigm that begins from the text to eventually assign meaning to it. That is, interpretation is an iterative process that will gradually add meaning to the analyzed text. In interpretive semantics, the meaning of language is composed of a structure built and integrated from linguistic data, intentions, and the interpreter's previous knowledge. Furthermore, the vision of global structured data, the text meaning, is achieved by projecting general linguistic data onto local semantic entities.

Currently, most of the text classification processes are based on statistical techniques. However, there are subtleties in the language that cannot be covered by statistical approaches. Our proposal consists of complementing the statistical methods using a method derived from interpretive semantics. Interpretive methods require a solid quantitative foundation, as we will see later.

Statistical methods have been seen to produce successful results when there is enough information available to train the algorithm; in these cases, the entities' semantic complexity is low. However, this does not work well when there is little information to train the algorithm, and they begin to show a deficiency as the semantic complexity of the elements grows.

In our approach, we see that semantic features can improve the efficiency and effectiveness of statistical methods. Using interpretive semantics, we show how to improve the statistical classification of texts. We take the interpretive semantics theory and build a computer tool to classify texts according to their semantic features.

4.2. Interpretive semantics

Text analyzers have tried to reduce the ambiguities of context by using statistical resources looking for textual cohesion. This is complex because a sentence's meaning is interpreted concerning the whole text and is not reduced to a succession of sentences. The interpretive elements provide textual cohesion facts independently of the syntactic structures and indifferent to the sentence's limit. The interpretive perspective tends to move away from structuralist models when dealing with the morpheme's semantic levels, the statement, and the text [Rastier-96].

A text alone does not contain what is required for its interpretation. The interpretation problems consist of the requisition of the necessary objects and the rejection of the idle ones. In this regard,

the meaning of the sentence influences the meaning of the words. To summarize, the sentence's form cannot be the continuation of the words that comprise it because its meaning depends on the sentence. Most of the classifiers that use NLP have focused on syntax, but this is not the case for semantics, partially because a sentence's meaning is not a function of the meaning of its parts.

Interpretive semantics is a system of descriptive forms with a semantic layer. It entails superficial relationships between the interpretive semantic elements. The meaning of a word is not itself the definition of the word. One interpretation is equivalent to one contextual semantic unit.

4.3. Structure of the interpretive semantics

The minimum unit of sense is universal and is called seme (SE)[2], the expression of the semes themselves is done through the use of the natural language, with no particular constrain. The union of two or more semes is named a sememe (S)[3].

A sememe is identified by a simple chain of characters, which permits identity and differentiation relations. The attribution of a seme to a set of sememes is known as taxeme (T). The taxemes only own contextual validity, and they work as an axis of interpretation. Thus, the taxeme is a relative element of the semantic approach that depends on the interpretive objects.

In interpretive semantics, an essential element is the isotopy (I), and is based on the redundancy of the information about a quantitative element. The traits of an isotopy are not directly observable since they are elements of the meaning. The quantitative feature is the base of its identification. The descriptive concepts of the interpretive semantics are not developed for a quantitative methodological model, but they offer several joints solutions to interpret the quantitative data [Pincemin-10].

The isotopy is used as a formal tool to capture objectives, is an independent structure, and is formed by a series of relations of identity between sememes, where the relations induce equivalent relations between sememes. However, it is necessary to roam between inferences to identify it.

Now, we will establish some limits and restrictions within the interpretive semantics framework that is necessary for our approach. The seme is a type of element similar to a set of sememes. A seme is a non-empty set of semes. The identity of the elements of S is intrinsic to this set. The

[2] [Tutescu-74] summarized the seme's idea thus: The minimum unit of meaning, the relevant feature of the semantic content, the invariant of meaning, is called the semantic mark, the semic marker, or seme.

[3] [Greimas-66] further defines the sememe as the effect of meaning between the combination of semes and the context.

cardinality of the set S is susceptible to evolution during interpretation: it remains, however, finite. E.g., "Musician," "Concert," "Concert-hall," "The-Royal-Albert-Hall" are sememes. Therefore, we describe the limits a taxeme T as

$$\forall t \in T, t \in \mathcal{P}(S), |T| \geq 2 \tag{4–1}$$

T is the set of taxemes, and its elements are annotated as t. T is a strict non-empty sub-set of all the parts of

$$S : (T \subset \mathcal{P}(S) \backslash \{\emptyset\}) \tag{4–2}$$

t is a non-empty set of elements of S, of cardinality $|t|$ higher than or equal to 2. The identity of two taxemes is based on the sememes that belong to them; it is an extensional identity, where

$$\forall t_1, t_2 \in T, t_1 \neq_s \ t_2 \Leftrightarrow (\exists s \in S \mid (s \in t_1 \wedge s \notin t_2) \vee (s \notin t_1 \wedge s \in t_2)) \tag{4–3}$$

in this case, the inequality refers to the sememe S in an extensional way. Moreover, we can define an identity between taxemes based on specific sememes:

$$\forall t_1, t_2 \in T, t_1 =_s t_s \Leftrightarrow \exists s \in S | s \in t, \wedge s \in t_2 \tag{4–4}$$

the isotopies are characterized by the following criteria: number of sememes, the distribution of these sememes, and the type of the isotopy. It is necessary to introduce another formal entity, the *speceme*. A speceme is an ordered pair of sememes that belong to the same taxeme formally

$$sp \in SP | sp = (s, s') \wedge \exists t_i \in T \ s \in t_i \wedge s \neq s' \tag{4–5}$$

the isotopic function I will be the one that assigns a seme to previously defined entities. I is a function of the following sets

$$I : SE \mapsto \mathcal{P}(SP) \times \mathcal{P}(T) \times \mathcal{P}(S) \tag{4–6}$$

if a seme is the function I, it will be called isotopic. Here, the next formula for an isotopy is arisen

$$I(se) = (\{sp_1, ..., sp_m, t_1, ..., t_m, s_1, ..., s_m\}) \tag{4–7}$$

alternatively, when going back only in the set S, it is got

$$I : SE \mapsto \mathcal{P}(\mathcal{S} \times \mathcal{S}) \times \mathcal{P}(\mathcal{S}) \times \mathcal{P}(\mathcal{S}) \tag{4–8}$$

with

$$\begin{aligned} I(se) = (&\{(s_{p1}, s'_{p1}), ..., (s_{pm}, s'_{pm})\}, \\ &\{\{s_1^{t_1}, ..., s_{m_{t1}}^{t_1}\}, ..., \{s_1^{t_1}, ..., s_{m_{tn}}^{t_n}\}\}, \\ &\{s_1, ..., s_p\}) \end{aligned} \tag{4–9}$$

[Tanguy-97a] highlights that the sense or notion of isotopy avoids the multiplication of notions of generic and specific. It includes the notion of Rastier's isotopy according to the principle of attribution of the assignation of the seme that is accompanied by a typology. Up to this point we have defined an elementary interpretive model with the basic elements to analyze the text at the primary level. We will use this model in the following and refer to it as interpretive semantics.

4.4. The interpretation process

[Rastier-96] defines the interpretative process as a series of operations for assigning meanings to a text. In our proposal to classify texts, we chose the semic analysis as a starting point, fundamental for the interpretative process. The semic analysis is a powerful tool that has been developed long ago, but no one has made a detailed description of the method.

The semic method is not limited to the study of verbal behavior. Intellectual acts or practical actions that involve no speech also have meaning and can be studied from their semic side. The semic analysis identifies the semes as elements of the meaning, finding clusters of the isotopies and their relations. The semic analysis is affected by diverse sources such as background, time, culture, or intellect. It looks simple for a human interpreter, but the diverse sources will mean that every interpreter will build its perspective and meaning according to its different situations. Appendix D

contains an extensive description of the semantic process. This section will limit the interpretation only to the analysis of relationships, concepts, individuals, and context.

4.5. Interpretive process proposal

We use the semic analysis of a text examined in the previous chapter as a basis for our interpretive semantics proposal. We add at the beginning of the process, a semic analysis of the text. Although this process avoids the direct inclusion of an interpreter, some verification tasks could suggest it. We modified the processes developed to allow vectorization tasks, metrics on semantic features, and NLP as the dependency analysis. The result of the process is an ontology, although the semantic analysis is more complicated than an ontology [Rastier-04a].

The rest of the chapter explains our approach with interpretive semantics, ontological forms, and the texts' interpretation. We establish a process composed of four stages: exploratory, structure, meaning, and verification.

4.5.1 Exploratory

The exploration and examination phase entails *taking a look* at the text before starting with the semic analysis. This helps the process optimize linguistic resources and to even foretell the result. Here, relying on the corpus's linguistics [Rastier-11] to classify and identify the themes where the text is more likely to belong. Techniques of textometry are used to find concepts and classify them to a theme. Finally, we recognize the named-entities. Thus, a sort of rustic isotopy is formed that will be later used to determine the contexts related to the text. This is a way to find out if the text belongs to the technical or literary form.

The text is a minimal linguistic unit, while the corpus is a whole in which this unit takes its meaning [Rastier-09]. This phase's primary aim has anticipated them and lined up corpus resources for the NLP models used in the following phases. Although this task is not thoroughly conducted, it is essential to this approach, where techniques of linguistics of corpus and *textometry* [Loiseau-18] are used to classify and obtain themes and *stylization*. Another objective is to restructure the text to simplify the analysis and discard the text in advance if the analysis is not possible, that is, in case there is no grammatical congruence. The final exploration is the complete result of the interpretative

analysis. This phase may be applied throughout the process. Since, frequently, there are analysis results that lead to improved exploration. The last phase of exploration is the complete result of the interpretive analysis. A text finds its sources in a corpus. Its production is based on this corpus and must be preserved or reformulated to be interpreted correctly.

4.5.2 Structure

This grammatical phase aims to find and create base structures like morphemes, lemmas, and lexemes. In this step, it is not possible to define a sememe. For this approach, only the nouns, verbs, adjectives, and adverbs are kept. The analysis of dependencies and co–relations defines the relations between the structures. This structure is generic or rustic in the sense that its elements lack meaning. Up to this point, they only show their existence in the text. Another critical part of this stage is that the grammatical category and the lemmatized form of the words in the different sections, chapters, paragraphs, and statements are defined here. Some grammatical structures are created and tagged using annotations to help us improve the next phase later in the process. Once the supporting corpus is understood, the initial analysis task must be addressed to evaluate each element that makes up the text.

4.5.3 Meaning

TABLE 4.1. THE SEMIC ANALYSIS.

Semes	Intensity	Glosses	Isotopies
Coca–Cola	afferent	Bebida gaseosa y refrescante, vendida a nivel mundial en más de doscientos países o territorios.	Liquido
bebida	afferent	Líquido que se bebe.	
gaseosa	inherent	Que se halla en estado de gas.	Estado
vendida	afferent	Traspasar a alguien por el precio convenido la propiedad de lo que se posee.	Soda
nivel mundial	afferent	El significado básico del sustantivo nivel es ”altura”, y en sentido figurado, categoría o rango.	Producto

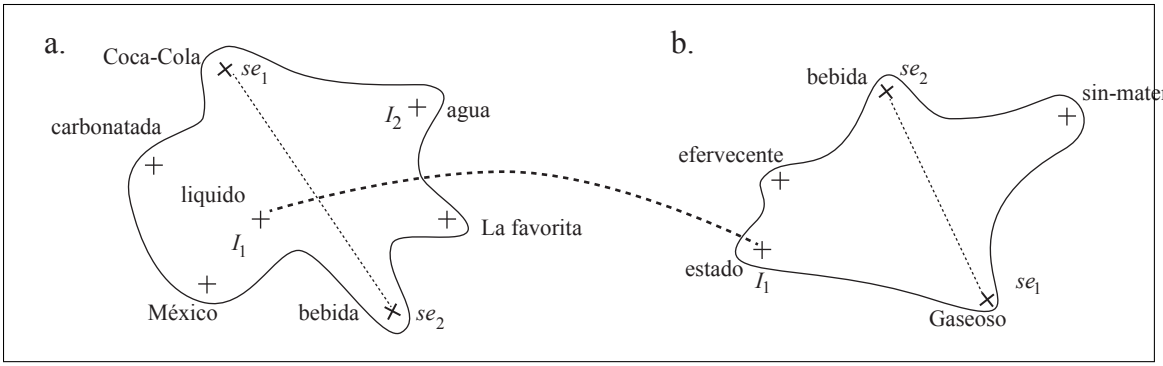


Fig. 4.1 Isotopy extraction, statistical models for the detection of isotopies in word vectors, contextual approach.

This phase provides the meaning of the structures previously formed. The elements are compiled with semantic traits. That means that every lemma is assigned with semantically complete words of the definitions of the sememes, considered as semantic characteristics. Here, the elements of the sources of definition, e.g., dictionaries, are *semantically stable*[4]. The semantic traits are clustered in morphological families, and a distribution of the semantic traits are established, forming preisotopies. Here, the semantic fields are vectorized, as well, to form a micro semantic surrounding. In the second part of this phase, the semantic network is created following Rastier's interpretative model. Tanguy's restrictions are applied to validate the structure, and an ontology of the interpretative model is created.

Each element will be provided with meaning. Here, reference sources will allow scanning for descriptions of each concept or individual. Compound elements such as */noun+noun/*, require special treatment. Their references or sources usually belong to a particular or specific domain. Identification of the interepretive intensity for the afferent semes is the consequence of a relation between two sememes in different taxemes, e.g., */weakness/* for "woman." Meanwhile, a default value has been assigned to each element in the inherent semes (that have not been defined as sememes, yet) the afferent intensity, and only the inherited occurrences are taken as inherent, e.g., */black/* for "raven." Table 4.1 shows the final assembly analysis. The construction of a preisotopy is intended for detecting the collective characteristics and traits of two or more terms, equation (4–6). It is a manageable way to begin the creation of the isotopy.

Using vector models we can extract a considerable number of features regarding the semes. In

[4] *Semantic stability* refers to the production of semantic traits.

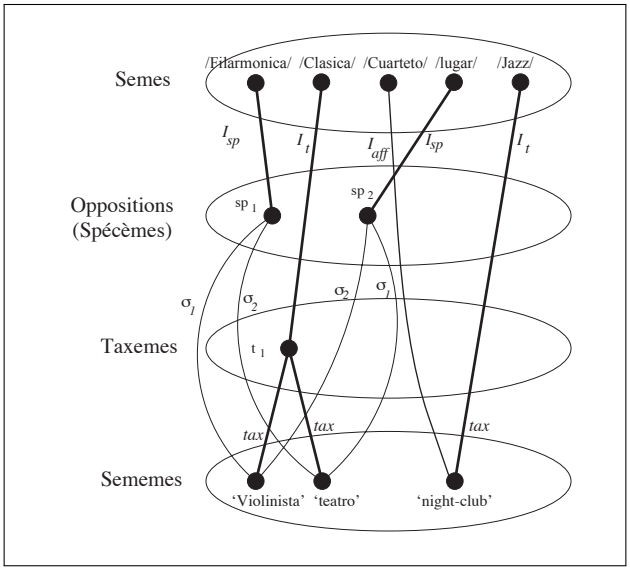


Fig. 4.2 Example of the interpretative semantics structure, adopted from [Tanguy-97a].

Fig. 4.1, we can see a sample with the most typical characteristics of the semes. Even though there are several terms, not all of them are good candidates, since they do not describe a meaningful association, e.g., the definition of "*Coca-Cola es una bebida gaseosa vendida a nivel mundial*," the semes "*Coca-cola*" and "*bebida*" where the first one is considered as an instance of "*Soda,*" easily detectable in the taxonomy of Wikipedia. Here, in the sample, it is seen that the only valid isotopy should be "*liquid*." However, other semes as "*agua*," "*carbonatada,*" are also representative and appear in the pattern. This is expected due to the nature of the vectors model formed by the repetition of words. To address this issue, a process of preisotopies suiting is based on the support of all the semes contained in a definition, where not only the semes will be analyzed. In the last example the two sememes "*liquid*" and "*Agua*," are comparing with "*gaseosa*" or "*vendida*" semes. However, initially, there is no success. This means that more elements of context are required to have a better interpretation of isotopies and semes. With them, it is found which isotopies have a better correspondence.

A. Interpretive semantics structure

The interpretive structure must begin to implement the restrictions previously defined in section 4.3. The purpose is to produce a similar structure to that proposed by Tanguy, as shown on Fig. 4.2, where the following operators have been defined: the group of sememes $(+S)$ must be constructed and associated with the taxemes $(+t, add(s,t))$, and the information about the relations of the sememe

IT WAS INEVITABLE: the **scent** of bitter almonds always **reminded** him of the fate of **unrequited** love. **Dr. Juvenal Urbino** noticed it as soon as he entered **the still darkened house** where he had hurried on an urgent call to attend a case that for him had lost all urgency many years be fore. The Antillean refugee **Jeremiah de Saint–Amour**, disabled war veteran, **photographer** of children, and his most **sympathetic opponent** in chess, had escaped the **torments of memory** with the aromatic fumes of gold cyanide.

Interpretation results:		Sample isotopy shapes:
Total semes:	8	####....# Solitude
Total sememes:	10	#..#....# Death
Initial classes:	8	###.....# Romance
New isotopies:	6	..#...#.. Friendship
Larges # of classes reached:	3	#.......# Sadness
Average # of classes/isotopy:	0.75	.... Hate

Fig. 4.3 Semic analysis on selected text *Love in the time of cholera*.

previously declared by the interpreter, with the activation of the function I.

In this stage, we managed the restrictions in the intersection and the inclusion of classes. To do so, and according to its initial proposals, the interpreter must choose between several possibilities. Therefore, this step will involve an operator of an election, selection, and operations related to sememes' movement respecting their taxemes $(sub(s,t))$.

The taxemes previously created and filled must be specified, configuring the order of their opposition graphs. Therefore, the operators should create *specemes* $(+sp)$ and activate them $(active)$. However, the scope does not allow us to work with restrictions and *specemes*. Fig. 4.3. shows the result of the interpretative analysis of the example definition that has been worked on in the semic analysis. The definitions have been written in prose. Besides, the essential features of the sub-features have been set. Section (a) shows a summary of the elements found in the analysis, while section (b) shows some samples of isotopies shapes.

The principle behind this transformation is the similarity between isotopies and semantic class whose distinction is only due to *specemes*[5]. The process of checking the correct semantic analysis entails comparing implementations in a manual and automated form. However, when comparing the results, the deficiencies in the automated process were remarkable. This was due to the automated

[5] [Tanguy-97a] defines a *speceme* as a support of the semes. It serves to specify which sememe a specific semema is assigned and differentiate it in the same taxeme.

process being more focused on structuring, whereas the manual one represents subjective visions of a text's meaning.

TABLE 4.2. COMPARISON BETWEEN TEXT STYLES.

Text	Precision	Recall	F–measure
Technical references	0.81	0.83	**0.87**
Literary novels	0.84	0.88	0.79
Social dialogues	0.53	0.57	0.59

An autonomous approach is highly dependant on semantic dictionaries. Table 4.2 shows the comparison results. Technical texts were taken from encyclopedia definitions for commercial products and services. Literary texts are fragments of literary works such as novels and tales. Social texts are conversations taken from social networks. These last texts do not include emojis nor photos. Given this, the significant meaning was lost and a low score; there was a significant loss of meaning and low scores.

B. Ontology transformation from interpretive semantics

The semantic analysis result is transformed into semantic forms, where the meaning ontology is kept with limited lexicometry. The lexical co-occurrences of words are qualified as semantic correlations and can be considered partial lexicalizations of a theme [Rastier-96].

With the construction of semantic forms, it is possible, in turn, to provide more meaning, adding semantic fields, to the units of the expression, including the ones of "*nivel-mundial*," such as the punctuation or the phonemes. At this point, the semantic transformation process is verified through the coherence in the resulting structures. The analytical and interpretative level of the generated semes of the previous phase must be checked. Notably, the different concepts are compared by ensuring the absence of contradictions within the same topic and uncover isotopies within the text. Connections between semantic features depend entirely on the co-occurrence; these connections pieces are using to create structures that enable testing the result of the interpretive analysis.

C. Transformations

Before setting up the rules of transformation, we will review Cimiano's proposal to to generate and populate an ontology. This will help us to establish the links between interpretive semantics and

ontology learning. This is the center of our proposal. We have an ontology as

$$\mathcal{O} := (\mathcal{C}, \leq, \mathcal{R}, \sigma_R, \leq_R, \mathcal{A}, \sigma_A, \mathcal{T}) \tag{4–10}$$

where $\mathcal{C}$ is a concept, $\mathcal{R}$ is a relation, $\leq$ is a sort. Moreover, there are two disjoint sets $\mathcal{A}$ and $\mathcal{T}$, whose elements are called relation identifiers, attribute identifiers, and data types, respectively. A semi-upper lattice $\leq c$ on $\mathcal{C}$ with top element root c called concept hierarchy or taxonomy, and finally $\mathcal{T}$ is a datatype. Now, the following rules were used to transform the semantic analysis into an ontology

1. Every isotopy is a concept whose relations must have at least two relations.
2. Semes and sememes represent concepts.
3. The taxemes are generalizations of a concept.
4. The attributes of the glosses are taken as objects or data attributes.

Each of these transformation rules has been assigned an operator to simplify the generation of the ontology. So, the operator $(+\mathcal{C}(add(se, s, t, es)))$ adds concepts. The *speceme* is considered as a concept class. However, it was not included in the implementation. Relations are marked by isotopies $(+\mathcal{R}(add(I)))$, and the most significant are encountered in indirect (inherent) semes $(+\mathcal{A}(add(se, s)))$ as well as types $(+\mathcal{T}(add(se, s)))$.

The result is a set $\mathcal{C}$ of concepts as

$$\mathcal{C} = \{\textit{Coca-Cola, bebida, gaseosa, nivel-mundial}\} \tag{4–11}$$

where "*Coca-Cola*" stands for a product instance. In the relations and attributes in the example ontology, there are the following signatures:

$$\begin{aligned} \sigma_R(\textit{vendida\_a}) &= (\textit{Coca-Cola}, \textit{nivel-mundial}) \\ \sigma_{\mathcal{A}}(\textit{azucar}) &= (\textit{Coca-Cola}, \text{contenido}) \\ \sigma_{\mathcal{A}}(\textit{gas}) &= (\textit{Coca-Cola}, \textit{contenido}) \end{aligned} \tag{4–12}$$

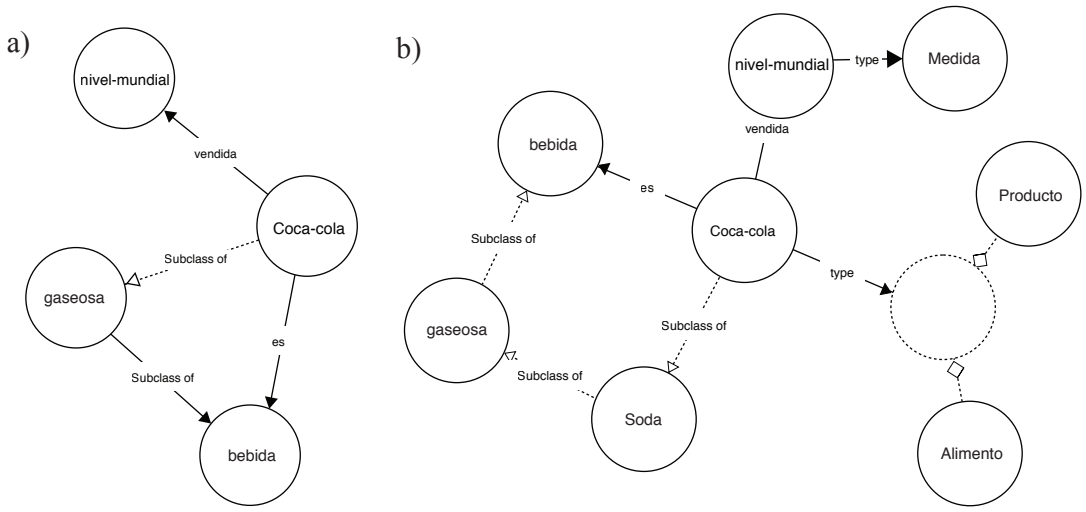


Fig. 4.4 Ontologies created with the transformation process, (a) based on SVM characteristics extraction, (b) created from IS feature extraction.

the following queries are used to validate the ontology and the last structure

$$\forall x(soda(x) \leftrightarrow \exists y\ soda\_of\ (x, y) \wedge \forall z(soda\_of\ (z, x) \rightarrow z = y)) \tag{4–13}$$

this first axiom states that every soda has a unique "*bebida*,"

$$\forall x(soda(x) \leftrightarrow \exists y\ soda\_of(x, y) \wedge bebida(y)) \tag{4–14}$$

while the second defines the concept of soda as equivalent to saying that there is a drink that stands in a $soda\_of$ relation with the corresponding "*bebida*."

4.5.4 Validation

Once the formalism is established, an ontology is created to regulate the construction of the ontologies of interpretation. The result in Fig. 4.4 compares the generated ontologies, (a) is an ontology formed with feature extraction applying simple quantitative methods. In this case, Word2vec[6] was used. While in (b), the extraction of features was using the interpretive semantic model. It was possible to have ontologies up to 20% more complex and descriptive. There is still

[6] Word2vec, *An efficient implementation of the continuous bag-of-words and skip-gram architectures for computing vector representations of words*, Nov. 10, 2020, https://code.google.com/archive/p/word2vec/.

low stability in the structure, especially if the stated conditions are breached. So, the tests were limited to simple statements.

TABLE 4.3. CLASSIFICATION PERFORMANCE

Group	Our proposal	Paralleldot	FastText
concept	0.79	0.92	0.82
individual	**0.68**	**0.32**	**0.21**
noun+noun	0.58	0.28	0.19
noun+verb+adj	0.48	0.10	0.07

Table 4.3 demonstrates that the semi-supervised classifiers as Paralleldot[7] and FastText[8] are more successful in classifying conceptual words, mainly because the training dataset has a large number of conceptual terms. However, when classifying words like individuals or compound data, the results were poor. Our proposed classifier has more certainty in classifying any words, even with words that were never trained.

The classification of concepts is the one that has the best results. The classification of individuals goes as low as 0.79; however, there are some critical ambiguity issues here, e.g., the term *Heineken* could be understood as a company, product, last name, or city. Thus, the classifier is limited to products and services, and heuristics were included to just filter these two groups. Compound concepts, those containing two or more nouns or formed out of verbs, are much more difficult to classify.

We found that immense complexity in classifying terms is in the task of semantic evidence extraction. that is, where the features of a term are searched. The space vector model does not take into account the language issues; that means they do not interpret the words' meaning. The relation is given statistically and, even though they have a good result, semantically speaking, there is no real understanding. So, complex terms do not have good results.

[7] Paralleldot, *Image recognition for perfect retail execution*, Oct. 28, 2020, https://www.paralleldots.com.
[8] FastText, *Library for efficient text classification and representation learning*, Oct. 28, 2020, https://fasttext.cc.

4.6. Future works

Although some primitive interpretative framework bases have been established along with this proposal, there have been several challenges, especially when implementing a computing tool based on interpretative semantics. Thus, there are several developments of computational models left to be solved: cover more classes for the specemes, the norms, the intensity of the semic elements and a new model for the extension identity.

4.7. Conclusions

In this chapter, we analyzed the formal theory of interpretative semantics using semantic techniques to interpret meaning. Unlike previous classification works, here, we have defined a process to extract features with more meaning. However, Rastier's theory, even though it sounds tempting, is entangled by several obstacles. We conclude that it is possible to arrive at a semantic interpretation using a computational model. Nevertheless, our model has some limitations, and we have not delved deep into the formalisms of interpretive semantics. Dealing with and solving these limitations will be left as future work.

5. IS-based ontology learning

A fundamental shortcoming of ontology learning, as presented in the previous chapter, is its inability to extract semantic traits over sufficiently long or syntactically complex texts. In a text, a sentence most often expresses several atomic facts is most reasonable when each of its elements expresses only a single, isolated fact. In this chapter, we construct a system for ontology learning and feature extraction, which will also serve to create an ontology with elements from semantic traits of the text. We will revisit the merging of this IS–based OL system into a QAS in the next chapter.

5.1. Introduction

OL is a technique from text meaning to detect patterns such as concepts and their relations and can be used to build from scratch or to populate previously created ontologies. Even though there are several methods and forms [Manning-08], [Gabor-16], and [Biemann-15] to create and populate ontologies, OL has a real benefit [Cimiano-06] since it has algorithms and tools to infer knowledge from specific text–domain collections automatically. It is common that OL approaches are used on ontology–based QAS because there is an improvement in the answer retrieval task [Unger-11]. A fundamental shortcoming of OL is its inability to extract semantic traits over long or syntactically complex texts.

Instead of relying on pure statistics, we would rather employ a hybrid learning approach using functionalities from statistical and linguistic ontology learning techniques guiding the extraction retrieval. At the base, we have a learning approach that can infer rules from text. We use an interpretive model that uses semantic fields and semantic traits as semantic evidence to build a model that learns to recognize good over harmful patterns in a text. In IS, transactions are defined in terms of words occurring together in a particular syntactic structure. The IS model is a structure of description forms with a semantic field as a whole system, structured along with simple relations between semantic units.

In this chapter, we use IS to enhance the feature extraction task in OL, in order to create an

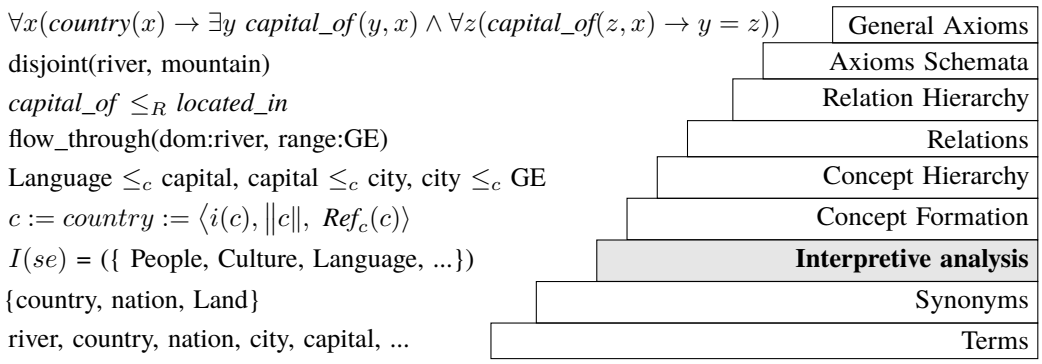


Fig. 5.1 Interpretive ontology learning layer cake.

ontology with elements from semantic traits of the text. We show that by using a IS model. One can effectively contribute to reduce the lexical gap compared to a baseline system where only known labels are used.

5.2. Ontology learning

Ontologies are a collection of information, taxonomies, and inference rules [Gruber-89]. Ontologies play a central function in our approach, as they constitute the conceptual basis of the domain where natural language expressions will be interpreted.

In this chapter, we will consider an approach to enhance ontology learning, specifically, one that enhances semantic traits retrieval. This method explicitly trains for the ability to learn new concepts, or it learns how to learn. While the concept of ontology learning is not new [Cimiano-06], [Gruber-95], [Völker-07], and [Faria-11], semantic models, along with techniques in deep learning, with increased computational power and large datasets, motivate us to revisit this approach in a new light. In Cimiano's OL approach, he defines tasks of text-transform in the ontology. In the transformation process of OL Fig. 5.1, we found that more and more tasks of logical operations are being added to the process. In the primary stages of extracting terms, synonyms and concepts are working by semi-statistical methods of natural language processing, word of vectors, or Wordnet with Synsets[1]. Relationships, concept, and relationship hierarchies align more with knowledge structure processes. Finally, there are the axiomatic logical operations.

[1] Sets of cognitive synonyms.

Formally, an ontology is defined as

$$\mathcal{O} := (\mathcal{C}, \leq, \mathcal{R}, \sigma_{\mathcal{R}}, \leq_{\mathcal{R}}, \mathcal{A}, \sigma_A, \mathcal{T}) \tag{5–1}$$

where, C is the set of concepts, R it is the set of relationships between ontology elements and its instances, A are attributes, and T types. Concept hierarchies or taxonomies are defined as a semi–upper lattice $\leq_{\mathcal{C}}$ on $\mathcal{C}$ with top element $root_{\mathcal{C}}$. A function $\sigma_{\mathcal{R}} : \mathcal{R} \to \mathcal{C}^+$ on $\mathcal{R}$ called relation signature. A function $\sigma_{\mathcal{A}} : \mathcal{A} \to \mathcal{C}$, called attribute signature, both functions work as unique identifiers inside an ontology. The semi-upper lattice $\leq_{\mathcal{C}}$ and/or $\leq_{\mathcal{R}}$ are defined as

$$\forall x \quad x \leq x (\textit{reflexive}) \tag{5–2}$$

$$\forall x \forall y (x \leq y \wedge y \leq x = y) (\textit{anti-symmetric}) \tag{5–3}$$

$$\forall x \forall y \forall z (x \leq y \wedge y \leq z \in x \leq z) (\textit{transitive}) \tag{5–4}$$

$$\forall x \quad x \leq top (\textit{top-element}) \tag{5–5}$$

$$\forall x \forall y \exists z (x \geq x \wedge z \geq y \geq \forall w (w \geq x \wedge w \geq y \to w > z)) (\textit{supremum}) \tag{5–6}$$

In this context of ontologies, we will refer to some elements (designed as *supremum*) as the least common subsume (LCS) where

$$LCS(a, b) = (Z | z \geq a \wedge z \geq b) \wedge (\forall w \notin > z (w \geq a \wedge w \geq b \in w \geq z)) \tag{5–7}$$

LCS of two concepts a and b it is the most specific concept, which is an ancestor of both a and b, where the concept tree is defined by the "is-a" relation Fig. 5.2. Often, we will call concept and relation identifiers just concepts and relations. Respectively, for simplicity a relation $r \in R$ with $|\sigma(r)| = 2$, we define its domain and range as:

$$\textit{dom}(r) = \pi_1(\sigma(r)) \tag{5–8}$$

$$\textit{range}(r) = \pi_2(\sigma(r)) \tag{5–9}$$

The extraction of terms is the initial task where we search for relevant S_C and S_R, where S is a term, provided by the lexicon to transform. The ontology domain is governed by the lexical elements

that make it up. Nevertheless, as did Cimiano in his initial approach we use synonyms in OL referenced from S_C and S_R, in a way that $ref_C(c)$, where ref is a function that receives synonyms from dictionaries, and increments the lexicon in the extraction and reinforces the learning of the characteristics and their relations. It is complex to establish a level of synonymy because of subtle differences in the words. At the same time, equivalents, e.g., in Wordnet [Miller-07] we show that the value of synset corresponds to this difference. To simplify this process, we take the synonymies as hyponymy phenomena as equivalences.

The objective of concept extraction task in OL is expressed as triplets, to form them we have $\langle i(c), [\![c]\!], ref_C(c)\rangle$ where $i(c)$ is the concept intension, this is a non-extensional definition of a particular concept and/or relation, $[\![c]\!]$ is the extension, and $ref_C(c)$ shows the realization in the corpus.

Although we know that the lexicon can contain complex structures impossible to model with triplets, we use an index defined in chapter 3 to evaluate the reliability of extracted triplet. The conceptual hierarchies are established between a set of concepts, where C, typically together with their lexical realization ref_C, learning pairs c_i, c_j, where $c_i, c_j \notin C$ such that $\leq_C = \cup X_{(i,j)}(c_i, c_j)$ forms a semi–upper lattice, e.g., *microprocessor, keyboard and hard disk*, in the concept hierarchy $\leq_C$ are depicted as *hardware*. We can refine the concept given a set of concepts C as well as a semi–upper lattice $\leq_C$ on C. The task here is to extend the existing concept hierarchy with additional sub–concepts of already existing concepts, thus refining the hierarchy.

Logical extensions and logical references of a concept c given its lexical reference function $ref_C(c)$, allow us to find new lexical realization s_i of the concept c, thus generating an extended $ref_C(c)$, e.g., $ref'_C(c) = ref_C(c) \cup X_i s_i$. As a result of lexical extension, we would, for example, add the term *processer* to the set $ref_C(computer)$.

5.2.1 Relations extracting

A binary relation restricts the learning of the relations; we can find concepts in C standing in some non–taxonomic ontological relation, in relation $r \notin R$, we can determine the right level of abstraction concerning the concept hierarchy for the domain and range of the relation. Lastly, we can get a hierarchical order over the relation in $\leq_R$.

5.2.2 General axioms

The $\mathcal{L}$–axiom system [Cimiano-06], is defined by concepts that we have, for example, disjointness or equivalence axioms, while for relations we have axioms describing the properties of the relation, e.g., transitivity and symmetry. The task here is to learn which concepts, relations, or pairs of concepts the axioms in our system apply to, e.g., we may want to learn which pairs of concepts are disjoint, which relations are symmetric, the minimal and maximal cardinality of a relation.

5.2.3 Population

An *instance_of* relation is the set membership relation between an instance $i \in I$ and the set $i_C(c)$ of some concept c, e.g., *instance_of*$(i, c) \leftrightarrow i \in I$. A similar description for the instantiation relationship: *instance_of*$_R((i_1, i_2, r \leftrightarrow (i_1, i_2 \in i_R(r))$. The tasks within ontology population are to learn *instance_of* an *instance_of*$_R$ relations.

5.3. Experimentation

5.3.1 Dictionaries

Through our experiments with the ontology learning process in Spanish, we aim to address two questions: (1) *Can our approach effectively learn from Spanish Resources?* (2) *What Spanish rules adaptations are required regarding English rules?* To answer these questions, we run our experiments collecting Spanish dictionaries to experiment in the OL process. We use DRAE[2] and Wikipedia[3] as Spanish dictionaries instead of LDOCE [Dolan-94] as used by Cimiano.

We compare three definitions: (1) LDOCE: launch *"a large usu. motor-driven boat used for carrying people on rivers, lakes, etc."* (2) DRAE: lancha *"Bote de vela y remo, propio para ayudar a transportar carga entre puntos cercanos de la costa."* (3) Wikipedia: *"Una lancha es una embarcación pequeña de vela para el transporte de personas."* (context: *"Lanchas"*). If we extract using Dolan approach the richest structures from dictionary entries, we have: (((CLASS BOAT/*lancha*) (PROPERTIES LARGE/*pequeña*)) (PURPOSE (PREDICATION (

[2]DRAE, Diccionario de la lengua española, Aug. 30 2020 https://dle.rae.es.
[3]Wikipedia, Wikipedia, Aug. 30 2020, https://www.wikipedia.org.

CLASS CARRY/*transporte*) (OBJECT PEOPLE/*personas*)))).

TABLE 5.1. EXTRACTION OF SEMEMES.

Measure	RAE	Disambiguation Wikipedia	Context
ubicación	1. f. Acción y efecto **de** ubicar	Localización geográfica es cualquier forma de localización en un contexto geográfico.	Geography
producto	1. m. Cosa producida.	Un servicio es un conjunto de actividades que buscan satisfacer las necesidades de un cliente.	Economy
servicio	1. m. Acción y efecto **de** servir	Similar	Economy
costos	2. m. Gasto realizado para la obtención o adquisicion costos	Similar	Economy
información	Acción y efecto **de** informar	Similar	no context
contacto	5. m. Relación o trato que se establece entre dos o más personas o entidades	Similar	Comercial
soporte	Apoyo o sostén	Similar	no context
plan	(1). (Del lat. plana). 1. f. llana (herramienta que usan los albañiles).	Similar	no context
saludo	1. m. Acción y efecto **de** saludar	Similar	no context

The above examples suggest that one can extract frame–based or feature structures from dictionaries, even in different languages, containing a wealth of semantic relations linking the different words together.

In several cases, DRAE is not a consistent structure. In Table 5.1, we observed that, in the case of Wikipedia, there is a more consistent corpus, the verb to-be is used most of the cases to define things, helpful to extract and determine context using machine-readable dictionaries (MRD).

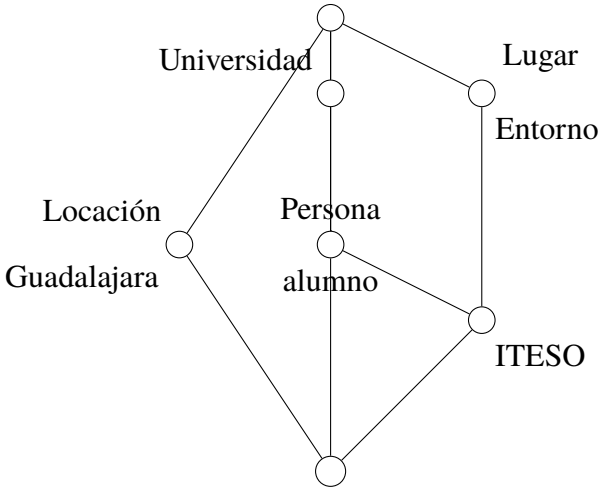


Fig. 5.2 Example of lattice generation.

Besides, LDOCE and DRAE are general dictionaries, free of context. These are the highest issues of ontology learning, that an ontology always reflects a way of conceptualizing the world or a given domain [Lehmann-15], and we needed more than one dictionary.

5.3.2 Extracting terms

We start extracting taxonomies using hypernymy and hyponymy relations from MRD, we are analyzing specific terms, named genius terms (GT). Forms of GT are "kind/a kind of," "sort/a sort of," "type/a type of" and taking the head of the NP following the preposition "of as" or "type" in the case of Spanish MRD. We have made an adjustment to include "is/are" and "have/has." Using these adjustments, we obtain better results with Wikipedia. The explicit definition or distributional hypothesis is important because we do not always have explicit information.

5.3.3 Formal concept analysis

We use FCA [Ignatov-14], [Bertet-17] and [Ganter-05] as the formalization method to extract concepts in IR. FCA is an applied branch of lattice theory that provides a model for describing a set of objects with a set of properties. FCA also elicits context, which may be used both to improve the retrieval of specific items from a text collection and to drive the mining of its contents. Our lattice is filled using conceptual exploration from Gamallo's approach [Gamallo-05], where a condition *cond* represents the set of linguistic properties that words (acquire combinations or

TABLE 5.2. COMPARISON OF DIFFERENT ONTOLOGY LEARNING APPROACH

OL Approach	SQuAD-es	Tourism	Our proposal
Cimiano_2010	-	48.82	-
Human score	89.23	-	98.1
Our proposal	**74.21**	-	**78.2**

TABLE 5.3. EXAMPLE OF FORMAL CONCEPT ANALYSIS CONTEXT

Concepts	*universidad*	*persona*	*lugar*	*instrumento*
ITESO	x	-	x	-
alumno	x	x	-	-
entorno	-	-	x	-
herramientas	-	-	-	x

requirements of nouns, verbs, and adjectives) must satisfy in order to be in position $\langle loc, w\rangle$. So, a linguistic requirement of w can be represented as the pair $\langle\langle loc, w\rangle, cond\rangle$, for a w and a specific description of a location *loc*, the paired block, *cond* represents a position concerning w. Nevertheless, conceptual exploration not only focuses on concepts, it can be used also to find meaningful attribute combinations [Ganter-16]. Using POS and dictionaries (optimized later as we will see in section 5.5) we only extracted noun–verbs/object pairs for the terms in Table 5.3, we used the conditional probability and semantic similarity (word2vec) to weight and dismiss the pairs with less significance. Fig. 5.2 shows an example of a lattice automatically derived from *"El ITESO busca que el alumno aprenda colectivamente, reflexione sobre el entorno social que lo rodea y adquiera las herramientas para transformarlo."* Finally, the lattice resulting from this, $(\mathcal{B}, \leq)$ is transformed and compacted into a partial order $(\mathcal{C}', \leq')$ which, is closer to a concept hierarchy in the traditional sense.

5.4. Evaluation

In order to evaluate our approach, we compare if the relations in the automatically learned ontology are correct. The resulting ontologies are compared in terms of the defined similarity measures, thus yielding the agreement of different subjects on the task of modeling an ontology. We compare our results in Table 5.2. The OL model by Cimiano compared with our proposal. To confirm that our dataset is clean, we add new results with human accuracy to transform in all the texts used in our custom datasets. Although we do not compare the same datasets, we try to show the percentage of concepts and relationships extracted by approach. We use three different datasets to evaluate the above algorithms. Tourism is a larger corpus merged with three corpora, Mecklenburg, Lonely Planet, BNC, contains 101,332 tokens and 6,971 documents. SQuAD [Rajpurkar-18] is a reading comprehension dataset consisting of 100,000+ questions posed by crowdworkers on a set of Wikipedia articles. We manually translated 192 SQuAD V2 examples from the categories of "Computer Complexity" to test our approach in Spanish. Finally, ChatOnto is a corpus built by us (handcrafted), containing 1000 questions-answers from pages that offer products or services. We have also put questions that have no answer as suggested in SQuAD .

To evaluate our SQuAD–es, we compare our proposal with human score (HS). In terms of *F-measure* our proposal get 74.23, in comparison with HS that gets 89.21. It can be observed that the HS approach should perform better, but some questions contain ambiguities, with no answers. The state-of-the-art best score, 89.14, is by Google AI language. The best *F-measure* for tourism dataset is 48.82. We do not have another point of reference, and there is sparse information to be able to reproduce Cimiano's approach. However, to see that in his experimentation, he achieves less than half of the transformed text. We have better results with our approach ChatOnto since the dataset is less complex (more natural and more transparent verbal relationships), 78.2. However, this experimentation has not found unanswered questions. Answering the initial questions, although on less scale, compared to the English language, if the resources exist (although SQuAD does not come in Spanish), at least to execute the Cimiano's model (but not reproduce it). In Table 5.2, it is shown that the results were higher than Cimiano's evaluations.

5.5. Ontology learning enhancement

TABLE 5.4. SEMIC ANALYSIS OF "WEB PAGE: ABOUT ITESO".

Semes	Intensity	Glosses
ITESO	*afferent*	El ITESO, **Universidad** Jesuita de **Guadalajara**, es una universidad privada ubicada en la Zona Metropolitana de **Guadalajara**, Jalisco, México, fundada en el año 1957.
Universidad	*afferent*	**Institución** de **enseñanza** superior que comprende diversas facultades, y que confiere los grados académicos correspondientes.
Jesuita	*inherent*	Se dice del **religioso** de la Compañía de Jesús, fundada por San Ignacio de Loyola.
Guadalajara	*afferent*	Guadalajara **es** una **ciudad** mexicana, **capital** y urbe más poblada del estado de Jalisco.
Fundada	*afferent*	**Establecido**, creado.
1957	*afferent*	**año** normal comenzado en martes.
Pertenece	*inherent*	Tocarle a alguien o **ser propia de él**, o serle debida.
universidades	*afferent*	**Institución** de **enseñanza** superior que comprende diversas facultades, y que confiere los grados académicos correspondientes.
Jesuitas	*afferent*	La **compañía** de Jesús orden **religiosa**

The process of ontology learning described in the previous section can produce excellent results on ontologies from the text, but with limitations, because it uses English dictionaries. We aim at producing better results using interpretive semantics–based ontology learning, looking for semantic traits omitted by retrieval tasks in early experimentation. We create classes with more symbolic elements and attributes, which facilitate the transformation into clean, compact, and meaningful ontologies. In our experiments, we find that the additional semantic traits contribute to around 33.0 of the *f1–score*, and this will increase substantially when using more contextualized traits. We enhance the OL model using the IS model. The IS is an iterative model witch process gradually and

El **ITESO** es la **Universidad Jesuita** de Guadalajara. Fue **fundado** en **1957** y pertenece al conjunto de más de 228 universidades jesuitas en el mundo. Comparte con ellas la tradición **educativa** de 450 años, históricamente ubicada en el centro del **pensamiento mundial** y reconocida por la formación de **líderes** en todos los campos de las **ciencias** y las **artes**.

```
Interpretation results:
Total semes:                      8
Total samemes:                   10
Initial classes:                  6
New isotopies:                    3
Larges # of classes reached:      3
Average # of classes/isotopy:  0.75
```

```
Sample isotopy shapes:
##.......... Educación
.#.......... Religión
.....#...... Epoca
..#......... Locación
####....#.## Intelecto
```

Fig. 5.3 Interpretative analysis resulting from *"About ITESO"* web page.

append more and more meaning to the analyzed text. The semic analysis [Rastier-96] is an essential task in IS. It consists in the transformation of the implicit meaning into explicit meaning that the IR model can represent by a generic trait. The isotopies are resulting elements of the Semic analysis, like FCA, but these are elements that have more context. Our proposal is based on building isotopies to find features with more traits and characteristics. To achieve this, we use the extraction method based on semantic traits developed in previous section 5.4, where we look for semantic evidence in semantic fields. Create ontologies to enricher their elements are the central challenge in ontology learning.

Our optimization model considers modifying $ref_C(c)$, without losing generality, to reach $ref_C(I)$. We replace the concepts by more complex structures that allow us to relate terms without losing meaning. To start, we formally defined the isotopy element as $I = SEP(SP)\times P(T)\times P(SE)$, where SP is a Speceme, T is a Taxeme and SE is a Sememe. The lexicon for an ontology is (S_c, S_R, S_a, ref_C, ref_R,ref_a), where S_c, S_R and S_a are concepts, relations and attributes, here a relation $ref_C \subseteq S_c\times C$ is called lexical reference in concept, $\forall t_1, t_2 \in T, t_1 \neq_s t_2 \Leftrightarrow (\exists s \in S \mid (s \in t_1 \wedge s \notin t_2) \vee (s \notin t_1 \wedge s \in t_2))$. Here, the taxemes are generated extracting the $root_C$. An element from the lattice, the taxemes are concepts of high hierarchy in S_c, and their identity is based in the sememes that belong to them. As an example, consider the following sentence, "*El ITESO es la Universidad Jesuita de Guadalajara. Fundado en 1957, pertenece al conjunto de universidades*

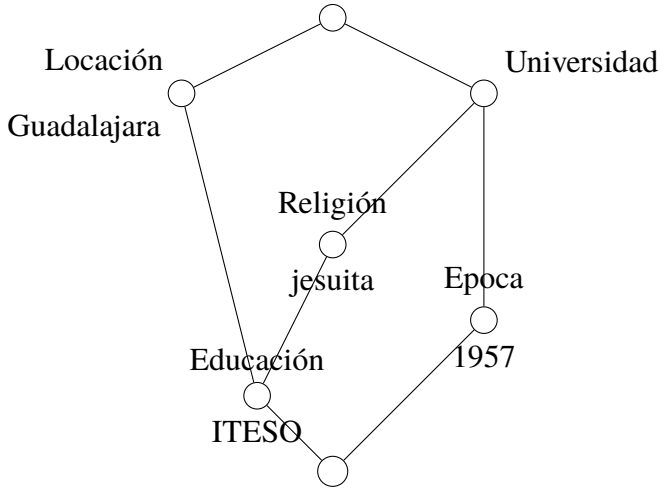


Fig. 5.4 Example of lattice automatically derived from *"About ITESO"* web page.

jesuitas en el mundo," triplets and attributes. Table 5.4 shows the isotopies found in the semic analysis that have been replaced in $root_C(I)$, Fig. 5.3. shows the interpretative analysis results used to generating a more concrete FCA, the final result is presented in the Fig. 5.4.

5.5.1 Learning enhancement results

TABLE 5.5. COMPARISON OF DIFFERENT ONTOLOGY LEARNING APPROACHES

OL approach	SQuAD-es	Tourism	Our proposal
Cimiano_2010	-	48.82	-
Human score	89.23	-	98.1
Our proposal	**74.21**	-	**78.2**
Our proposal+IS	**82.1**	-	**81.2**

The qualitative results, shown in Table 5.5 shows that the learned model is better than our last approach, we find that the additional semantic traits contribute to around 33.0 of the *f1–score*.

5.6. Conclusions

In this chapter, we formally define an ontology for OL. We explore and suggest diverse tasks for IR in OL to enhance the concept and relation learning, as well as the ontology populations tasks. The evaluation measures provide in our experimentation show important improvements versus the original approach. The main contribution of this chapter is the combination of OL with IS. Future research should thus further examine complex relationships between concepts and relations in the form of rules or axioms.

6. Question and answering systems

The objective of this chapter is to show in a brief historical framework the evolution of the QAS, mainly ontology-based QAS. Also, the elements that form a QAS are presented, how the interaction between each of them works, here some IR models and methodologies are detailed. Finally, we review the last advances in this field and review the most significant obstacles for the QAS, we propose solutions and put ourselves into experimentation.

Question answering systems are promising approaches for accessing knowledge. They offer people an intuitive way, instead of complex and structured query languages. The natural language–based question and answering systems have two primary challenges when applied to the real–world. The first is to understand questions as requirements and their structural complexity. The second is to produce answers with accuracy and confidence. Ontologies can increase the quality of the answers produced, providing a formalization of necessary knowledge for interpretation, and support for the construction of meaning representations.

Moreover, the more semantic information the system uses, the better the precision and correctness of the answer it achieves. However, this requires a new interpretative process in a particular information retrieval task to enhance the extracted characteristics from text and produce ontologies with more productive classes and relations. In this chapter, we review the main characteristics of the question and answering systems based on ontologies. We review advantages, difficulties, and challenges. Finally, we conclude discussing future directions in Question and Answering Systems research.

6.1. Introduction

Nowadays, the volume of information that a person requires to make decisions is higher than in the past. People wonder about personal, professional, and day to day inquiries or want to confirm something they do not remember. The Internet has expanded into an almost unlimited information base. Every time that a question is asked, there is more than one source of information to rely on to analyze it. There is too much information to understand, and there is limited time to get answers.

Furthermore, it is necessary to confirm that what is learned has reliable bases.

The artificial intelligence community, researches and develops tools to simplify the access to knowledge for ordinary people. Here, the IPA were created just after the middle of the last century [Simmons-67]; since then, their use has been growing exponentially. Mainly due to the emergence of personal smart devices. According to the world economic forum, by 2020 there will be more than 50 billion devices connected to the planet[1]. The development of tools to manage extensive information enabled the applications to use higher knowledge. It contributed to making the devices smarter [Russell-10], as well as the improvement of internet broadband and the lowered services costs in the cloud made the Internet affordable for everyone.

IPAs can interact with people in several forms, with commands, alerts, suggestions, queries, and information filters. IPAs operate with question-answer and can enhance the human–computer interaction, significantly establishing complete dialogues with the persons [Dourish-14]. A QAS is an IPA that extracts answers, among different data sources, to natural language questions in a range of topics and parses them to machine language. QAS represents some significant challenges since the comprehension of natural language is required. Most of QAS extract factoids answers to address the issues through clues or lexical answer type; afterward, they search for words that are classified and pondered using statistical techniques. The QAS process works fine when there is a lot of information available to learn from it. Statistical methods undoubtedly remain an essential process to extract information. Nevertheless, statistical techniques need to be complemented by semantic techniques. This work has turned to semantics to enhance the purely quantitative approach in the information retrieval process. There are several techniques to extract answers in the QASs. Proposals of QAS based on knowledge have had successful results [Ferrucci-12], [Xu-14], [Baudis-15], [Kuznetsov-16], [Hakimov-17] and [Diefenbach-18]. The release of ontological standards such as OWL provided by the W3C has accelerated their use. Using OWL, the QAS benefit, as this standard offers symbolic knowledge representation and inference rely on ontologies. Moreover, works like [Dubey-18] and [Lopez-10] confirmed that the representation of the knowledge increases the QAS score significantly.

There are structures as semantic networks, triplets, n–dimensions, semantic–graphs, or semantic

[1] AI-Revolution, *The AI revolution is coming fast*, Feb. 9, 2019, https://www.weforum.org/agenda/2016/08/the-digital-revolution-is-here-but-without-a-revolution-in-trust-it-will-fail.

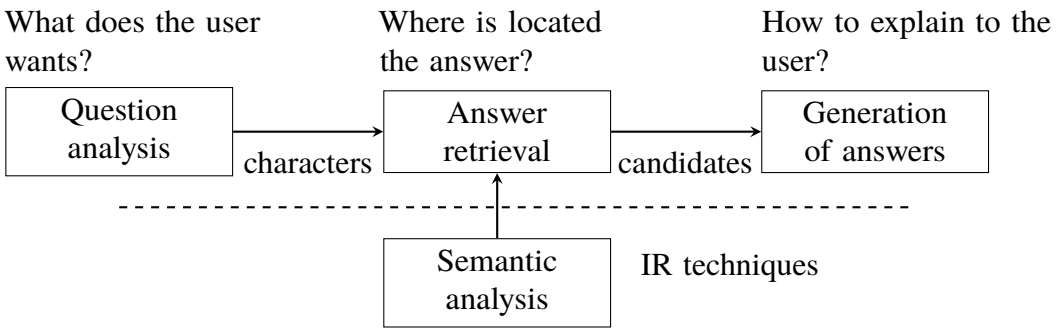


Fig. 6.1 QAS: The three basic stages and the interaction with the semantic analysis.

boxes. Notwithstanding, it is questionable that they do not have enough expressiveness as the ontologies do [Rastier-04]. The ontologies are the best way to maintaining a linguistic structure, given that defining their concepts and their relations is far from representing the language. An important application of ontologies as formal representations is to provide semantics (meaning) and domain knowledge to the data.

Formally a QAS is given as a question $q \in Q$, where we assume that an RDF search engine already provides us a set of relevant triplets $T = \{t_1, ..., t_n | t_i \in T\}$ Each triplet t_i consists of few elements, we define two finite integers $s, e \in [0, L]$ to represent the starting and ending position of the answer, respectively. Here L denotes the total number of tokens of the given element set. Fig.6.1 shows the main stages in a QAS.

6.2. The questions answering systems

According to the state-of-the-art, the QAS are classified into three groups: based on statistics, based on knowledge, and a combination of both, which is a hybrid approach. In a statistical approach, it is frequent that the extraction of information is done with text mining techniques. The sources of data are usually raw, as corpuses and unstructured texts in natural language. In this approach, the sources of community questions have started to be taken more seriously. Here deep learning techniques can provide successful results [Rao-16]. The approaches based on knowledge obtain their answers using previously structured resources, like databases, graphs, or ontologies. The responses are given through concepts and their relations. The deductive efficiency is high because they use SPARQL and descriptive logic reasoners. This last group uses all the techniques that are available to obtain the answers, but, in the end, it uses only the ones that have a higher score. This group gets

focused on a logic frame to find the best candidate answers. We are interested in two types of QASs that are described below.

6.2.1 Retrieval base QAS

In these QAS, several techniques of machine learning are used to extract similar questions for similar answers that had been before asked by people, e.g., forums like Quora[2]. This approach is less complex (in comparison to logical approaches) than other approaches to release and support. It is efficient and reliable in its answers as long as they are being used. In contrast, they are recurrently supervised, demand vast volumes of tagged data, and required robust hardware.

6.2.2 Knowledge base QASs

TABLE 6.1. COMPARATIVE OF THE QUESTION AND ANSWERING SYSTEMS

Stage	(1) Statistical	(2) Knowledge	Hybrid
IR	Text mining	Triplets mapping	Both (1,2)
Source	Raw	Triplets storages	Both (1,2)
Focus	Quantity	Concepts and relations	Logics meaning
Approaches	Machine learning	Sparql, Tableaux reasoners	YodaQA
Precision(f1)	30–40%	32–60%	40–65%

Although knowledge base QASs are using statistics as a base, the primary method to obtain answers with more reliability is using graph structures to interpret, deduce, and reason answers. Their reasoning is based on the meaning of the words, concepts, and relations involved in the question. That is, the whole meaning of the question is obtained before the QAS starts to seek for the answer. In this same way, the answer will be processed semantically. e.g., *When was the Foundation of Guadalajara?* It is equal to ask: *How old is the Second Most Important City in Mexico?* The state-of-the-art of the Knowledge QAS uses a base of semantic structures such as pre–processed ontologies,

[2]Quora. (2019, Feb. 8). *Quora: Home* [Online]. https://www.quora.com/.

e.g., Freebase[3], Quora, Yahoo Questions[4], or Stack Overflow[5]. In a controlled environment, this approach will succeed when there is a considerable amount of transformed knowledge housed in semantic structures. However, the systems that generate knowledge on the fly are required to generate and validate structures of knowledge dynamically.

It was found, therefore, that the relation question–answer–source is given in base of question–forms. Even though there could be more combinations, the proposed model only considers five simple forms of questions known as factoid, list, hypothetical, confirmation, and causative. In contrast, the Internet has abundant raw data that has not been translated into graphs, and there are few types of researches on how to use question statements in a graph structure, not only as a SPARQL query.

In Table 6.1, it can be observed that the precision and recall of a question–answer gets improved according to the type of analysis that the QAS uses. There is a remarkable improvement when better interpretation is obtained from the question and the sources. Therefore, it can be stated that computational semantics can improve reliability since semantic evidence is found.

6.2.3 The role of the semantics

Formal logic has benefited the linguistic analysis [Pereira-82]; however, several issues have come up because of their nature, e.g., generality and ambiguity. Some studies, such as Interpretive Semantics [Rastier-96], could lessen or support these logic models. Nevertheless, it is quite early, and a more in–depth study is required to justify their use.

The first QAS [Simmons-67] were created with few statistical information. Back in those days, there were not any public internet sites regarding questions and answers available as there are nowadays; that is why the comprehension of the question was focused more as a matter of grammar analysis of the question. Today we have numerous corpuses. Ancora [Taule-08] is a Spanish corpus that contains more than 500,000 lemmas and morphological categories, constituencies, and syntactic functions, verbal semantic classes, named entities, and co–reference relations.

These resources help in breaking down and facilitating a better understanding of the grammar

[3]Freebase. (2019, Jan. 27) *Freebase API,* Feb. 8, 2019, https://developers.google.com/freebase/.

[4]Yahoo Answers, *Yahoo Answers*, Feb. 8, 2019, https://answers.yahoo.com/.

[5]Stack Overflow, *Stack Overflow - Where developers learn, share, build careers*, Feb. 8, 2019, https://stackoverflow.com/.

structure of the language. However, it does not make the computers understand the meaning of what is said. Semantics study the meaning of the words and sentences [Chomsky-57], [Melcuk-12]. There are several theories about the semantics of natural language and how to treat them [Tarski-38], [Partee-76] to mention some of them, In Montague's publication [Partee-76] it was feasible to use semantic as computing model to parse from natural language to logic, even in a direct form, something that later was not practical because of the complexity of the language [Blackburn-05].

Our work takes the QAS community advances from recent years. It uses them in combination with computational semantics techniques in order to make the QAS interpret, not only the quantitative knowledge but also the meaning of the questions. By doing so, the QAS has more probabilities of producing the best answer. We propose to create a hybrid QAS that uses techniques of information retrieval and knowledge bases to produce answers to questions that have never been asked, at least, that are not stored in a question–answer dataset.

6.2.4 Computational semantic analysis

Linguistics has been used in computer science to analyze. Since then, there have been several advances in the linguistic field, particularly in the development of tools for grammatical analysis, such as part of speech [Toutanova-03], co–reference relations, or dependency parsers that have been developed with success. However, semantic analysis has not been developed in the same manner. Some authors state that it is due to the complexity and that today symbolical semantics are not mature enough. In order to benefit from the advantages of different analyses in semantics, the research has been aligned with symbolic semantics from a computational point of view. Some important works in this area include Semantics of the Truth, proposed in Tarski's [Dubey-18], and later continued by his disciple Montague. Chomsky's influence is also notorious in most of the works about Semantics. Melcuk [Melcuk-12] provides a vast theoretical panorama. Rastier adds his proposal on interpretive semantics.

6.3. Components of questions answering systems

QAS has been evolving and improving their assertiveness with the time, with better results in statistical techniques and methods. Identifying their components and taking advantage of what

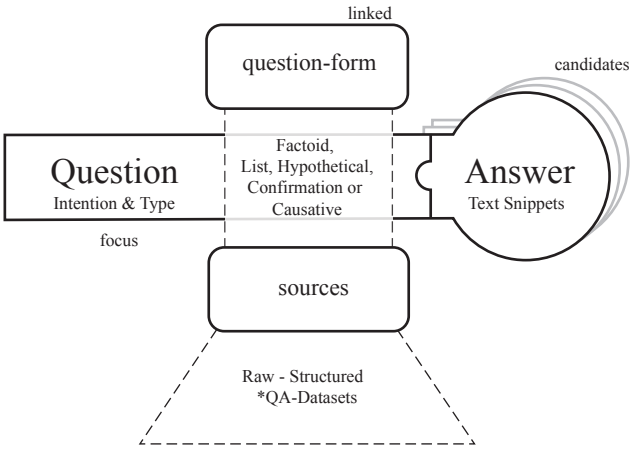


Fig. 6.2 QAS: Generic model proposal.

have worked might help in speeding up their development and improve them. Working with natural language makes QAS be complex systems, and human language is expressive, ambiguous, and implicit. QAS has been a matter of interest for quite a long time. There is a broad spectrum of approaches that have been developed in the last decades. QAS tends to divide its processes into three big stages: where the question is asked, the answer is found, and finally, the user's answer is expressed. There is no standard model that defines the characteristics or limits of every component. Most of them are custom implementations that depend on particular situations.

Fig.6.2 shows our approach that contains two essential components, the first analyzes the question to extract requirements that a person has regarding specific information whose response is unknown. Here, questions are classified according to their intention and assessing questions difficulty. We use mostly context–free grammars, to structure the question sentences, and interpretive semantics to analyze the questions and semantic evidence to validate. The second component uses the clues obtained from the analysis of the question and recognizes candidate answers in data sources. Some logical methods are used to reason and to produce an answer taking into account the predefined form of the question text, e.g., indirect, direct, open–close, mix, and other types of questions. Lastly, we build an answer with a similar form to the initial question to clarify the answer to the user.

6.3.1 Analysis of the question

We can classify questions according to their type of answer, most of the question in QAS are factoids, that means, they expect facts as the answer. Less frequent are the QAS that work with

questions of lists types, where the answers are sets of entities, can contain factoids, e.g., *"Which are the most developed European countries?"* there are also self–confirmation questions, that is, the question includes the answer, with some information to be confirmed: *"The first president of America was Washington, right?"*. The causal questions *"how"* or *"why,"* e.g., *"How is the traffic in the new airport?"* and lastly, hypothetical or pragmatic questions, e.g., *"What would it happen if...?"* The result of the analysis has a high impact in the assertiveness of the answer of a QAS since a more significant amount of candidate answers of high quality can be obtained.

6.3.2 Answer selection

Here is where the QAS is connected to information sources to find patterns, preferably with semantic evidence, where there are question–answer relations. The first stage of the analysis begins with the quest of patterns of the "*WH*" kind, e.g., *why, who, what,* or *where*. Most of the systems use simple heuristics, also known as "*WH*" [Dayal-16] , to determine the type of question. Thus, to classify if the question is about a person, location, time, or a combination of the previously mentioned. The work of [Li-06] about the classification of the questions through learning is an example of how to analyze questions. However, it is not just about classifying the type of questions, it may work in simple questions, but when the demand for information is involved, the complexity is determined by the level of detail and the implicit or explicit the information is required.

It is necessary to produce answers, identify where to examine, which are the sources of information with more possibilities to locate data linked to what was asked. So, the first step is to define the type of question. That is, what is it being looked for? if it is a fact, e.g., *"Where was Albert Einstein born?"* or *"Where would the Olympic Games of 2020 be held?"* or if it is a thing, e.g., *"What is a bread toaster?"* or if a person is just validating information that is known by *"him/her"* but is not entirely sure that is right, e.g., *"Was Obama the 44th President of the U.S.A?"* If these examples were analyzed, it could be said that the question (1) looks for a date, a person, and an event, while the question (2) looks for a location, an event, and an event on a given date. With this information, now, it is possible to determine the position of the source of text, if it is related to the question, as well as the percentage of probabilities that there is to determine and evaluate if it is necessary to analyze more deeply. The tag questions have more chances of getting a more successful result since they do not require an accurate answer. It is just a matter of comparing if the critical components

of the question that are in the text. In most cases, it is not necessary to have an in–depth analysis. However, if there are some implicit elements, it would be necessary to rephrase to extract more specific elements.

Regarding the sources of information, for this work, they were reduced to a few paragraphs in order to reduce the complexity of the study. So, the text analysis is made with a more specific effort in statistical, grammar, and mainly semantic techniques.

Thus, two techniques were defined to complete the tasks of determination/measuring of the answer: Dimensional and Structural approaches: (1) Featural dimension. Once it was determined that there is a chance of answer, the text entities, also known as proxies, can be measured with the features of the question. The method intentionally formulates a fictitious answer based on the question, so it has a comparing unit and applies metrics to determine the closest text to the question. Here, vector space models are used. (2) Structural. Similar to the previous method, if there is a good diagnosis about the possibility of finding an answer, a semantic graph of the text is generated, which is compared with a semantic graph previously generated from the question. Here, Ontology matching techniques are used [Euzenat-13] to determine parts of the graph that have higher coincidences.

TABLE 6.2. QUESTION TYPES FROM TEXT

Question	*Type FACTOID*
¿Quién es Juvenal Urbino?	Person
¿Quién es fotógrafo?	Person
¿Quién es doctor?	Person
¿Quién juega ajedrez?	Person
¿Quién muere?	Person
¿Qué había dejado de ser urgente?	Thing
¿De qué se trata el texto?	Theme

6.3.3 Answers production

Once the system finds a set of candidate answers with a high score of possibility to be an answer, a process to parse to natural language begins. A simple and understandable response is generated according to the question-form that the user asked. It is known that the answer is linked not only to the question but also to the kind of question. Table 7.2 shows the relationship between question and answer.

6.4. Some remarks

Up to this point, we described a QAS approach that combines different techniques to address the question answering problem. In the first stage, we review an information retrieval engine working on different knowledge bases to extract relevant candidate contexts for each question and candidate answer pair. In the second stage, we examined models based on ontologies to analyze each triplet to analyze questions and extract more candidate answers. We conclude with the statement that the more semantic information the system applies, the more reliable the precision and correctness of the answer QAS achieves.

6.5. Improving question answering systems using interpretive semantics

In previous chapter 5, we analyzed the meaning of semantic fields in chatbot dialogs (question–answering). We use an interpretive semantics model in combination with ontology learning techniques to transform text passages, questions, and answers into ontologies, assertion, and technical boxes. In our experimentation, we obtained a better *f–score* compared with simple statistical approaches trained on texts with sophisticated styles. This section is an extension of this latter approach. We propose to generate alignment with the questions. Then we create a semantic structure with them. We take into consideration particular characteristics of interrogative clauses such as question type, intention, nested, subordinate or dependent clauses, and question difficulty. We link components to connect the natural language. Surprisingly, little work has been done on the effect of context in question answering and supporting dialogs. Indeed, previous questions and answers will affect the

answer selection.

6.5.1 Creating knowledge–based QAS

Several QAS approaches transform question sentences into query sentences to get information from isolated sources. [Bird-09] uses context–free grammar to construct a basic ensemble of SQL sentences and execute queries on databases. In [Cimiano-09], [Hakimov-17], Cimiano uses the ontology lexicon model Lemon[6] to build SPARQL queries, which he used in his approach of ontology–based question answering DUDES[7]. In Baudi's [Baudis-15] QAS approach, he assembles a set of SPARQL using fuzzy logic methods [Zhang-14] to enhance the candidate answers from Wikidata[8]. Baudis' approach is based on the wiki info–boxes and is highly dependent on them. This hinders the adaptation of his approach to other open domains of raw text. Recently, one of the successful approaches in the transformation of queries is Berant's proposal [Berant-14]. Using probabilistic combinatory categorial grammar [Kwiatkowski-10], it generates an ensemble of meaning representations in formal logic sentences paraphrasing the original question. In his last approach SEMPRE[9], he shows the potential of ontological queries using scratching.

Transforming questions into SPARQL in the previous approaches are generally treated as an atomic task since question answering is not treated as a conversation. In a dialog flow, there are related questions, also known as highly compositional or sequential question answering. These have been studied by [Iyyer-17] and [Pasupat-15] as a semantic parsing problem. When we analyze sequential questions on a dialog, we examine isolated questions. Therefore, it is difficult to reach the correct answer, and it is hard to handle a second turn with a follow-up question.

A. Question classification and intention detection.

The segmentation phase is an initial preparation phase in our approach, as we have suggested in previous works [Kalyanpur-16]. In our approach, we use a segmenter for the Spanish proposed

[6]LEMON. The lemon cookbook, 20 Aug. 2019, https://lemon-model.net/lemon-cookbook.pdf.
[7]DUDES, Lightweight implementation of DUDES, 20 Aug. 2019, https://github.com/ag-sc/DUDES.
[8]Wikidata, The free knowledge base, 14 Aug. 2019, https://www.wikidata.org.
[9]SEMPRE, Semantic parsing with execution, 15 Aug. 2019, https://nlp.stanford.edu/software/sempre.

by Da Cunha [Da-Cunha-12]. DiSeg[10] is a sentence splitter that applies syntactic rules to insert segment boundaries into the sentences. Da Cunha's model facilitates the inclusion of rules. This is very convenient because we can add and combine semantic features into the model rules.

We classify text to assign a question to an appropriate category. In order to understand the question to a level that allows determining some of the constraints, the question imposes on a possible answer. There are several approaches, from applying *wh* heuristics to lexical answer type (LAT). Mohasseb [Mohasseb-18] proposes a question classification framework based on their grammatical structure. He identifies different patterns and uses machine learning algorithms to classify them. However, the grammatical structure lacks semantic analysis. Huang [Huang-15] and Li [Li-06] proposals use a learning question classifier in order to evaluate semantic features. They replace each word by its semantic class in the given context. These last approaches are based in Wordnet[11] for semantic analysis and classification. Nevertheless, they do not use context in the classification analysis producing many misinterpretations in complex texts such as novels.

Our approach combines the methods mentioned above and selects semantic classes with context for most of the semantic information sources to enlarge the coverage of the question recognizer. We define a question classification to be a multi–class classification task that seeks a mapping from an instance to one of the classes. This classification provides a semantic constraint on the sought-after answer.

After question classification, like Watson [Lally-16] and [Kalyanpur-16], our QAS implements a first component inspired in YodaQA[12]. Our LAT processes and assigns certain types to each question.

B. Semantic treatment for finding important traits.

One way to get a root element of a sentence is to analyze its grammar dependencies. We can accomplish this using the dependency parsing method proposed in chapter 5. We re-train the oracle dependency, so it uses the resulting traits from the semic analysis. In the previous section, we find that IS can bring better characteristics to IR tasks. We extend our semantic treatment algorithm and tasks for this purpose, adding simple heuristics. Nouns and verbs (in pure form) are transformed into

[10]DiSeg, *A discourse segmenter for Spanish*, 21, Aug. 2019, http://dev.termwatch.es/esj/DiSeg/WebDiSeg.
[11]Wordnet, *A Lexical Database for English*, 21, Aug. 2019, https://wordnet.princeton.edu.
[12]YodaQA, *Open source question answering system*, 21, Aug. 2019, http://ailao.eu/yodaqa.

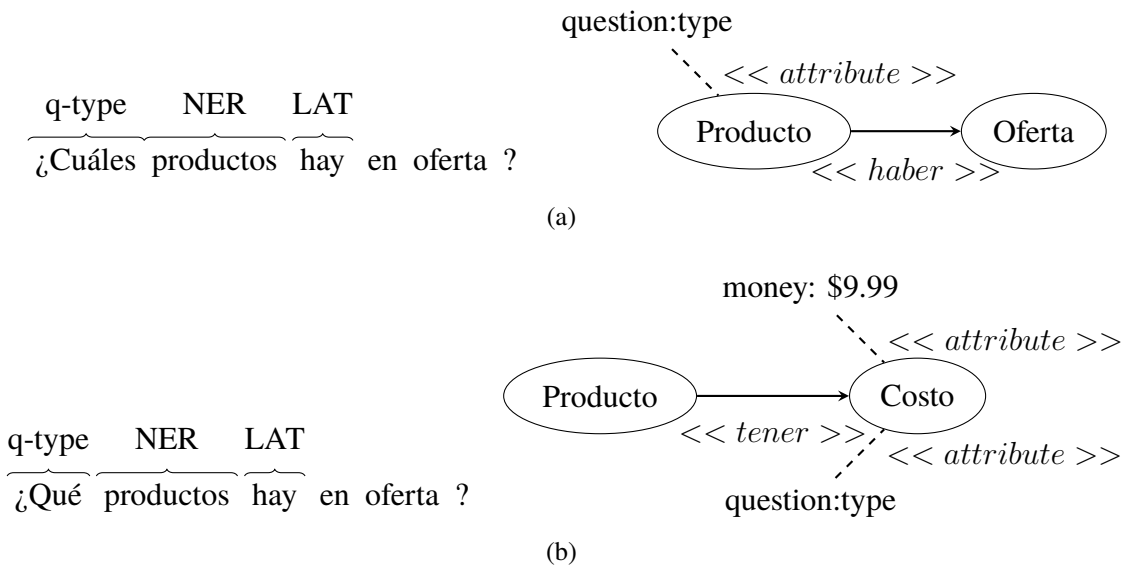


Fig. 6.3 Question–transformation where the partial ontology it is associated with SPARQL.

classes, the adjectives and adverbs are assigned as attributes, and simple prepositions, conjunctions, and interjections are treated as connectors.

As in chapter 5, we use a semi–upper lattice to generate graph structures. Then we validate them using semantic evidence. We measure using characteristics such as similarity meaning. If the expressivity of the text is reduced (less than 0), we refine the resulting graph by measuring meaning and similarity over the terms existing in the graph. We proposed a method to increase the expressivity in the elements by locating some explicit evidence and converting it. This process employs knowledge from external resources, which is semantically relevant to resolve ambiguities.

C. Question and context

Modeling conversations is a challenge, as the questions can be highly context-dependent, elliptical, and even unanswerable. Recently, almost all the language models embed context into a sequence of vectors [Huang-18], [Chen-17]. The context in machine comprehension has been successfully extended because there have been large datasets available. Typically, machine comprehension methods use attention to focus on a small portion of the context and summarize it with a fixed-size vector, couple attentions temporally [Seo-16] and [Levelt-82].

Like Chen [Huang-18], in each question, we compute attention in the word level to enhance context word embeddings with the question. Each question at the *i-th* turn is embedded into a sequence of vectors, where i is the maximum number of questions in the conversation. For each point in the passage, our model matches the context of this point against the encoded question from multiple perspectives and produces a matching vector. Initially, we had considered a supporting memory, but it is no longer necessary if we keep in mind the context of the dialogue. Furthermore, the ambiguity increases significantly when the context in the questions is removed.

Another interesting baseline is [Wang-16], a multi-perspective context matching (MPCM) model that given a pair of questions and passage. The MPCM model estimates probability distributions through the following six layers.

6.5.2 Connecting question–answering

Correctly answering questions requires recognizing objects, attributes, and relationships, as well as counting, performing logical inference, making comparisons, and leveraging common sense world knowledge. In our approach, every question is transformed into a graph, yielding the answer to the question, in order to enable rich analysis that would be impossible by dealing directly with natural-language questions.

A. Finding question correspondences with formal queries

We first evaluate the performance on each question type, defined as the outermost function in the model. (1) We check the existence questions, determining whether a specific type of object is present in the complete graph. (2) We compare attribute questions, through a series of compound queries, we determine whether two objects have the same value for some attribute. In Fig. 6.3, we show a sample of questions transformation. (*a*) shows the first question and main, (*b*) is the second question, and depends on (*a*). Once a natural language question has been mapped to a formal query, its answer can be retrieved by executing the query on a semantic graph.

B. Finding question correspondences with ontology matching

Ontology matching [Euzenat-13], [Arnold-13] aims at finding correspondences between semantically related entities of different ontologies. In our case, we have two ontologies: the question and the data ontologies. We use RiMOM [Li-09], a dynamic multi-strategy ontology alignment framework.

This system estimates similar characteristics for each parallel task. These characteristics are used for dynamically selecting and combining two matching methods; linguistic similarity and structural similarity implemented as three similarity propagation strategies such as concept–to–concept, property–to–property, and concept–to–property.

6.6. Experimentation

Initially, we evaluate our approaches using conversational machine datasets [Rajpurkar-18] and [Reddy-18], however, our model is designed for Spanish and although the datasets can be translated from English to Spanish the outcome is an incomplete decontextualized text, due to the complexity of the style in the articles. Also, the datasets are created for statistical tests; each paragraph or context has several questions through the statements (question-answering pairs). If the statistical approach answers a question, the algorithm locates the most similar question and obtains the corresponding answer. On the other hand, our approach constructs an answer.

We assemble a dataset that combines texts from three different domains, such as literature, middle school history, and online drug catalogs. Several crowd workers crafted these questions so that they were relevant to QAS. The new dataset contains a plausible answer–something of the same type as what the question asks for.

a. The SEP dataset was created from 5th–year history books issued annually by the Mexican Ministry of Public Education (SEP in Spanish). It contains around 10 contexts, 50 paragraphs, and more than 500 question–answering with inferences. b. The PRODH is a catalog of common drugs with descriptions and recommendations for use in English and Spanish. This dataset consists of 100 contexts and more than 1000 questions answers. c. LATS is the third dataset. It is a compendium of fragments of Latin American literature works (which have been translated from Spanish to English), this data set contains 40 contexts and more than 100 peer questions.

Table 6.3 contains samples of the elements of the built dataset. Table 6.4 summarizes the datasets

TABLE 6.3. EXAMPLES OF PARAGRAPHS IN OUR DATASETS

Source	Context	Example
SEP_1	Entre 1870 y 1920, nuestro país vivió cambios muy importantes. Con la llegada de Porfirio Díaz a la presidencia se inició la recuperación de la economía y la reconciliación entre los grupos que hasta entonces se disputaban el poder; con nuevos préstamos se restructuró la deuda externa y otros países invirtieron en México.	**Question**: "¿Cuántas décadas duró el Porfiriato? **Prediction**: "Entre 1870 y 1920", **Answer**: "3 décadas"
PROCH_1	NIMESULIDA tabletas está indicado como coadyuvante para el alivio de la inflamación, dolor y fiebre producida por infecciones agudas de las vías respiratorias superiores. Dismenorrea primaria, inflamación, reumatismo, esguinces, torceduras, fracturas, artritis reumatoide, osteoartritis, bursitis, en intervenciones quirúrgicas, tromboflebitis y desórdenes ginecológicos.	**Question**: "¿Que es la nimesulida?" **Prediction**: "coadyuvante para el alivio de la inflamación", **Answer**: "anti-inflamación"
LATN_1	Era inevitable: el olor de las almendras amargas le recordaba siempre el destino de los amores contrariados. El doctor Juvenal Urbino lo percibió desde que entró en la casa todavía en penumbras, adonde había acudido de urgencia a ocuparse de un caso que para él habla dejado de ser urgente desde hacía muchos años. El refugiado antillano Jeremiah de Saint-Amour, inválido de guerra, fotógrafo de niños y su adversario de ajedrez más compasivo, se había puesto a salvo de los tormentos de la memoria con un sahumerio de cianuro de oro.	**Question**: "¿Quién es doctor? ¿Quién juega ajedrez?¿Quién muere?¿Qué había dejado de ser urgente?¿De qué se trata el texto?" **Prediction**: "Juvenal Urbino, Jeremiah de Saint-Amour", **Answer**: "Juvenal Urbino"

TABLE 6.4. DATASET ELEMENTS DESCRIPTION

Elements	Datasets		
	SEP	PROCH	LATN
Context	50	50	100
Pharagraphs	50	50	100
Questions	200	92	1200
Answers	200	92	1200
Unanswerable questions	0	5	50

created. Our approach was compared to BIDAF[13] that is a bi-directional attention flow network, and a hierarchical multi–stage architecture to model representations of the context paragraph at different levels of granularity [Seo-16], and Allen[14] [Gardner-18] that is a framework for applying deep learning methods to natural language processing research.

6.7. Results

Although Neural Network-based models are trained with broad information, we notice that our model obtains the most reliable results in the test performed. This is mostly due to the use of reasoners. We observe that the analysis of the rich semantic structures has generated compact ontologies that facilitate the treatment of the questions and obtain more accurate answers. Also, it should be noticed that our algorithms reason even after having located answers so they can perform operations to improve responses, for example, calculate dates, synonyms, or root elements. We implement this last functionality inspired in the Berant's [Berant-14] work.

Although in Table 6.5, it is observed that we are still far from reaching human analysis, we see that the increase in our *f1–score* is significant compared to the other statistical approaches. The complexity, particularly of the narrative form in the analysis, is evident in our results. LATN is a challenge for any analyzer and even more difficult for statistically unsupervised models. Our

[13]BIDAF, *BiDAF Demo,* 21, Aug. 2019,http://allgood.cs.washington.edu:1995/
[14]Allen, *AllenNLP - Demo*, 21, Aug. 2019, https://demo.allennlp.org/reading-comprehension

TABLE 6.5. QUESTION AND ANSWER RESULTS BY CLASS.

QAS	Datasets		
	SEP	PROCH	LATN
Human performance	97.0	93.0	100
BIDAF++	50.0	24.0	12.5
AllenNLP	25.0	12.0	–
Our approach (en)	**–**	**45.5**	**36.0**
Our approach (sp)	**73.0**	**41.0**	**31.0**

$f1$ reached 31% and 36%. In conclusion, this score was only possible by applying the reasoning mechanisms of our analyzer. Moreover, the reasoners only work correctly if structured information in ontologies is semantically rich.

6.8. Future work

Future work should aim to improve QAS. (1) Semantic questioning theories such as Elliott's [Elliott-17] can help to deepen further in the analysis of the meaning of the questions in a QAS. (2) A synthetic question–answering generator over texts. (3) Extend datasets in Spanish texts for tests. (4) A method for measuring complex questions. Our emphasis is on complex questions that refer to multiple entities and relationships [Lu-19]. (5) Detects unanswerable questions with a method to determine if a correct answer is not present (or it is not stated in the context) in the corpus and stopping the process. (6) An unfolding process simplifies complex sentences, identifies the subordinate clauses, the active and passive voice.

6.9. Conclusion

In this report, we introduce a modular question answering system that can be used to test interpretive semantics models. We emphasize formal semantics and employ only a minimal amount of hand-crafted heuristics. At this point, our approach is already demonstrating a reasonable factoid question answering performance. It can answer set dialog questions correctly, to find over half of

the questions in the top five answers, and to consider the correct answer for just about 36% *f1–score* on our datasets. In many cases, especially in new texts, we could only locate correct answers using logical inferences. The semantic analysis gives advantages in the QAS because it provides more meaning in their features and enhances the location of correct answers. Finally, we observed that dialog context is crucial to understand and answer questions. We also hope our work helps to clarify how IS contributions are essential to a minimal working modern Question Answering System.

Conclusiones generales

Los enfoques estadísticos mejoraron con éxito en los últimos años debido a que cada día hay más información etiquetada disponible y al incremento de la capacidad de procesamiento. Sin embargo, aún estamos lejos de tener una solución completa para los QASs. Los sistemas inteligentes requieren componentes más efectivos que razonen e infieran. No es suficiente con los métodos estadísticos y probabilísticos del análisis del texto para obtener una interpretación completa, es decir, necesitamos que los sistemas también descifren el significado para que interpreten. En este trabajo, presentamos propuestas que ayudan a inferir y razonar aportando más significado a la extracción del texto. Como examinamos en el transcurso de este trabajo, las ontologías y la semántica interpretativa son elementos que pueden mejorar el análisis computacional (automático) del lenguaje. Dado que las ontologías pueden generalizar sobre su estructura y la semántica interpretativa va más del conocimiento enciclopédico de las palabras hasta la interpretación del sentido.

Nuestro enfoque combina métodos de aprendizaje ontológico con la semántica interpretativa y mejora la extracción de características especificas en el texto. La semántica interpretativa, como vimos en este trabajo, es prometedora para la recuperación de información. También experimentamos con un QAS basado en lógica descriptiva para determinar si un hecho de consulta está relacionado con algún hecho en el texto donde se busca. Además, demostramos que podemos bajar la complejidad (computacional) del formalismo lógico, manteniendo muchos de los beneficios de tener una lógica subyacente fuerte.

En el capítulo 1, exploramos los formalismos lógicos para realizar inferencias directamente en la forma superficial del texto. Las ventajas de este formalismo nos dan una notación lógica de validación en nuestro análisis semántico.

En el capítulo 2, desarrollamos una propuesta para transformar facturas digitales a redes semánticas. La transformación de facturas electrónicas a ontologías fue exitosa con nuestra estrategia de mapéo. Creamos una ontología robusta y compacta que permite componentes semánticos, como los razonadores, de una manera muy simple y eficiente. Nuestro prototipo mapeo con éxito la información de una facturas electrónica a una estructura semántica en OWL. Los resultados pudieron ser

verificados a través de consultas SPARQL y DL. Consideramos que el resultado fue exitoso dado que hubo poca o nula pérdida de información durante la transformación. Adicionalmente, el prototipo tuvo un rendimiento óptimo para los requisitos de hardware y software usados. Este proceso de transformación es importante porque mostramos como explotar el conocimiento almacenado en cadenas de texto crudas proporcionando características semánticas para su interpretación

En el capítulo 3, presentamos un modelo lineal elemental. Encontramos que las distancias entre palabras encontradas en un vector espacial para la relación en un texto determinado pueden crear un efecto empírico de evidencia semántica que se requiere para tener una medida de similitud. Este efecto, aunque es adecuado, carece de un análisis formal de ambos términos. Establecimos el vector espacial como elemento esencial para encontrar mayor evidencia semántica y generar proxies con significado semántico rico. De esta forma mejoramos las tareas de generación de ontologías. En este capítulo, demostramos que es posible clasificar palabras utilizando medidas semánticas.

En el capítulo 4, establecimos bases para un marco de semántica interpretativa. Utilizamos la teoría formal de la semántica interpretativa para la interpretación del significado en el texto. A diferencia de otros trabajos de clasificación, aquí, encontramos una forma de extraer características con un contenido más interpretativo. Presentamos un método para obtener significado a través de un vector espacial como recurso literario. Nuestra propuesta hace posible llegar a una interpretación inicial.

En el capítulo 5, definimos formalmente la ontología para el aprendizaje ontológico. Exploramos varias tareas de recuperación de información sobre el aprendizaje ontológico, para mejorar la obtención del concepto, sus relaciones de aprendizaje y poblar una ontología. Las medidas de evaluación proporcionadas en nuestra experimentación tienen mejoras importantes de acuerdo con el enfoque original. La principal contribución de este capítulo es la combinación del aprendizaje ontológico con la semántica interpretativa.

En el capítulo 6, propusimos un enfoque de QAS que combina diferentes técnicas semánticas para mejorar su score. En la primera etapa, revisamos un motor de recuperación de información que trabaja en diferentes bases de conocimiento para extraer contextos candidatos relevantes de cada par de preguntas y respuestas. En la segunda etapa, examinamos modelos basados en ontologías para analizar cada tripleta, analizar preguntas y extraer respuestas candidatas mas significativas. Finalmente, concluimos con la afirmación que, a más información semántica usada por el sistema,

mejor será la precisión y corrección de la respuesta que un QAS obtiene.

Esta tesis deja posibilidades de futuros trabajos de investigación.

El proceso de clasificación semántico propuesto en este trabajo puede ser complementado y tener una mejora notoria sí se le incorpora un proceso de identificación de palabras compuestas, esto amplía el alcance de la clasificación y reduce la perdida de información.

Trabajos futuros pueden continuar implementado el modelo de semántica interpretativa con los operadores faltantes de la teoría de Rastier y ampliar la cobertura del análisis del texto. En este trabajo, sólo usamos los elementos esenciales, pero la teoría de la semántica interpretativa es extensa y propone muchos más elementos, como los especemas, el multi-núcleo, las normas y la intensidad de los elementos sémicos.

Desarrollar un modelo de identidad extensional, identidades entre dos conceptos. Llamadas extensionales porque solo dependen del contexto para ser interpretado. Esto debe aumentar el resultado de las relaciones de un concepto. Además, trabajos futuros deberían examinar más a fondo las relaciones complejas entre conceptos y relaciones en forma de reglas y axiomas.

Con respecto a los QASs, crear un modelo usando las teorías de cuestionamiento semántico de Elliott [Elliott-17], para profundizar más en el análisis del significado sobre las preguntas.

Elaborar un método de medición de preguntas complejas. Las oraciones complejas o compuestas se refieren a múltiples entidades y sus relaciones. Reducir oraciones complejas, identificar cláusulas subordinadas, la voz activa y pasiva, son tareas que facilitarían el análisis interpretativo.

Desarrollar una estrategia para detectar preguntas incontestables, sin respuesta, y evitar la búsqueda de un elemento no presente en el corpus. Esto se podría hacer calculando profundidad de las relaciones y el anidamiento de las relaciones.

El aprendizaje automático de la interpretación del conocimiento esta lejos de concluir. Este trabajo y su estudio sobre modelos semánticos contribuyen al uso significado de los textos, como características primitivas fundamentales para mejora de los sistemas inteligentes.

Hemos explorado la semántico desde diferentes perspectivas; similitudes, campos semánticos, evidencia semántica y estructura ontológica. Encontramos que la interpretación va más allá del texto y/o del interprete. Esperamos que esta disertación pueda inspirar la investigación en la dirección de la semántica interpretativa y el uso de ontologías como base en los QASs para razonar e inferir conocimiento.

General conclusions

Statistical approaches have been successfully improved in recent years since more labeled information is available every day and increased processing capacity. However, we are still far from having a complete solution for QASs. Intelligent systems require more effective components that reason and infer. It is not enough with the text analysis's statistical and probabilistic methods to obtain a complete interpretation. That is to say, and we need that the systems also decipher the meaning to interpret. In this work, we present proposals that help infer and reason by providing more meaning to the text's extraction. As we examine in this paper's course, ontologies and interpretive semantics can improve computational analysis of language. Since ontologies can generalize about their structure and interpretive semantics goes more from an encyclopedic knowledge of words to interpretation of meaning.

Our approach combines ontological learning methods with interpretive semantics and improves the text's extraction of specific features. Interpretive semantic, as we saw in this work, is promising for information retrieval. We also experimented with a descriptive logic-based QAS to determine if a query fact is related to some fact in the text where it is searched. Together, these contributions have created a question-answering system that uses description logic to determine any fact entails a query in the corpus. Furthermore, we have shown that it is possible to relax the strictness of logical formalism while still maintaining beneficial properties of underlying solid logic.

In Chapter 1, we explored logical formalisms to make inferences directly from the surface form of text. The advantages of this formalism give us a logical validation notation in our semantic analysis.

In Chapter 2, we develop a proposal to transform digital invoices into semantic networks. The transformation from electronic invoices to ontologies was successful with our mapping strategy. We create a robust and compact ontology that allows semantic components, such as reasoners, efficient and straightforward. Our prototype successfully mapped the information from an electronic invoice to a semantic structure in OWL. The results could be verified through SPARQL, and DL queries. We consider that the result was successful since there was little or no information loss during the

transformation. Additionally, the prototype performed optimally for the hardware and software requirements used. This transformation process was necessary because we showed how to exploit the knowledge stored in raw text strings by providing semantic characteristics for its interpretation.

In Chapter 3, we presented an elementary linear model. We found that the distances between words found in a spatial vector for the relationship in a given text can create an effect of semantic evidence required to measure similarity. This effect, although adequate, lacks a formal analysis of both terms. We established that the space vector is an essential element for finding more semantic evidence and generating proxies with rich semantic meaning. In this way, we improve the ontology generation tasks. In this chapter, we show that it is possible to classify words using semantic measures.

In Chapter 4, we established foundations for interpretive semantics framework. We use the formal theory of interpretive semantics for the interpretation of meaning in the text. Unlike other classification works, here we find a way to extract characteristics with a more interpretive content. We present a method to obtain meaning through a space vector as a literary resource. Our proposal makes it possible to reach an initial interpretation.

In Chapter 5, we formally defined ontology for ontological learning. We explore various information retrieval tasks on ontological learning to improve the obtainment of the concept, learning relationships, and filling an ontology. The evaluation measures provided in our experimentation have significant improvements in line with the original approach. The main contribution of this chapter is the combination of ontological learning with interpretive semantics.

In Chapter 6, we proposed a QAS approach that combines different semantic techniques to improve its score. In the first stage, we review an information retrieval engine that works on different knowledge bases to extract relevant candidate contexts from each pair of candidate questions and answers. In the second stage, we examine ontology-based models to analyze each triplet, analyze questions, and extract a more significant number of candidate responses. Finally, we conclude that the more semantic information the system uses, the better precision and correctness are achieved by the QAS.

This thesis leaves possibilities for future research work. The semantic classification process proposed in this work can be complementary and have a noticeable improvement if identifying compound words is incorporated; this broadens the scope of the classification and reduces the loss

of information.

Future work can implement the interpretive semantics model with the missing operators of Rastier's theory and expand the text analysis coverage. In this work, we mainly use the essential elements, but the theory of interpretive semantics is extensive and proposes many more elements, such as the specemas, the multi-core, the norms, and the semic elements' intensity.

Develop an extensional identity model, identities between two concepts. Extensional calls because they only depend on the context to be interpreted. This should increase the outcome of a concept's relationships. Furthermore, future work should further examine the complex relationships between concepts and relationships in the form of rules and axioms.

Concerning QASs, create a model using Elliott's semantic questioning theories [Elliott-17] to go deeper into analyzing the questions' meaning.

Develop a method for measuring complex questions. Complex or compound sentences name multiple entities and their relationships. Reducing complex sentences, identifying subordinate clauses, and active and passive voice can facilitate interpretive analysis.

Develop a strategy to detect unanswerable questions without answers and avoid searching for an element not present in the corpus. This could be done by calculating the depth of the relationships and the nesting of the relationships.

Automatic learning of knowledge interpretation is far from over. However, this work and its study on semantic models contribute to the meaningful use of texts as fundamental primitive characteristics for improving intelligent systems.

We have explored semantics from different perspectives; similarities, semantic fields, semantic evidence, and ontological structure. We find that the interpretation goes beyond the text and the interpreter. We hope that this dissertation can inspire research in the direction of interpretive semantics and ontologies as a basis in QASs to reason and infer knowledge.

Appendix

A. List of internal research reports

1) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Improving retrieval-based question answering," Internal Report *PhDEngScITESO-19-17-R*, ITESO, Tlaquepaque, Mexico, May 2019.

2) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Towards an ontology-based question answering architecture," Internal Report *PhDEngScITESO-19-17-R,* ITESO, Tlaquepaque, Mexico, May 2019.

3) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "QAS: interpretative semantics base ontology learning," Internal Report *PhDEngScITESO-19-17-R*, ITESO, Tlaquepaque, Mexico, May 2019.

4) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Using interpretive semantics to enhance ontology learning method," Internal Report *PhDEngScITESO-19-17-R*, ITESO, Tlaquepaque, Mexico, Jul 2019.

5) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Logical and grammar treatment of the meaning of the text for information retrieval," Internal Report *PhDEngScITESO-19-17-R*, ITESO, Tlaquepaque, Mexico, May 2019.

6) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Using interpretative semantics to enhance text classification," Internal Report *PhDEngScITESO-18-17-R*, ITESO, Tlaquepaque, Mexico, May 2018.

7) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Semantic text classification: algorithms," Internal Report *PhDEngScITESO-18-17-R*, ITESO, Tlaquepaque, Mexico, May 2018.

8) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Semantic text classification: implementation," Internal Report *PhDEngScITESO-18-17-R*, ITESO, Tlaquepaque, Mexico, May 2018.

9) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Semantic text classification: a quantitative approach," Internal Report *PhDEngScITESO-18-11-R*, ITESO, Tlaquepaque, Mexico, May 2018.

10) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Automatic transformation of digital invoices to semantic networks," Internal Report *PhDEngScITESO-17-37-R*, ITESO, Tlaquepaque, Mexico, Nov. 2017.

11) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Semantic Processing," Internal Report *PhDEngScITESO-17-09-R*, ITESO, Tlaquepaque, Mexico, Jan. 2017.

12) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Semantic web: an overview of the state of the art," Internal Report *PhDEngScITESO-15-21-R*, ITESO, Tlaquepaque, Mexico, Dec. 2015.

13) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Towards the semantic web," Internal Report *PhDEngScITESO-14-22-R*, ITESO, Tlaquepaque, Mexico, Dec. 2014.

B. List of publications

B.1 JOURNAL PAPERS

1) L. M. Escobar-Vega, V. Zaldívar-Carrillo, and I. Villalon-Turrubiates, "Semantic invoice processing," *Journal of Intelligent Fuzzy Systems,* vol. 34, no. 5, pp. 2913-2922, 2018.

B.2 CONFERENCE PAPERS

1) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "*Semantic invoice processing,*" in *International Symposium on Language Knowledge Engineering (LKE-2017)*, Puebla, Mexico, Nov. 2017, 5. 2913-2922.

B.3 CHAPTER IN A BOOK

1) L. M. Escobar-Vega, V. H. Zaldivar-Carrillo, and I. Villalon-Turrubiates, "Comparative analysis and implementation of semantic-based classifiers," in *Advances in Artificial Intelligence, 17th Mexican Int. Conf. on Artificial Intelligence (MICAI-2018)*, G.Sidorov, O.Herrera-Alcántara, Guadalajara, Mexico, pp. 80–91.

C. Examples of words used in the classification

. EXAMPLES OF WORDS USED IN THE CLASSIFICATION

#	Term	Features	Complexity
1	agua	líquido, hielo, nieve, vapor, elemento, hidrógeno, oxígeno, común, supervivencia.	low
2	vino	bebida, uva, fermentación, azúcares, etanol, fruta, vinícola, vid, italia, francia, españa	low
3	café	bebida, granos tostados, cafeto, cafeína, sin alcohol, negro, leche, crema, chocolate	low
4	balón	pelota, juegos, terapéuticos	low
5	silla	asiento, respaldo, patas, personal	low
6	soda	bebida, agua, gaseosa, carbónico	low
7	sidral mundet	gaseosa, carbonatada, manzana, coca-cola	low
8	aspirina	ácido acetilsalicílico, salicilatos, medicamento, dolor, analgésico, antipirético, antiinflamatorio	low
9	heineken	pilsener, cerveza, alcohol, heineken	high
10	manzanita sol	manzana, pepsico, méxico, botellas, refresco, saborizado	low
11	tecate	municipio, baja california, pueblo mágico, turística	high
13	coca-cola	bebida gaseosa, refrescante, john pemberton, medicinal, rojo, estilizada botella	low

D. Semic analysis

Semic analysis is performed on a semiotic act, e.g., a text, by identifying the semes, that is, the elements of meaning (parts of the signified), defining clusters of semes (isotopies and molecules) and determining the relations between the clusters, relations of presupposition, comparison or between the isotopies Hebert-11. The development of semic analysis has been studied in both linguistics and computational semantics. Greimas, in his work of structural semantics Greimas-66, develops a comparative analysis between semes and lexemes to find generic isotopies. Rastier's studies of Interpretative Semantics Rastier-96 exemplify the development of a semic analysis. It also suggests a more in-depth analysis in Hebert-11, where he adds the statutes of intensity afferent and inherent of the same. Hebert Hebert-11, following Rastier's works, proposes a method based on three phases of analysis: exploratory, analytical, and comprehensive. The exploratory phase picks out the semes or isotopies present in the text or formulates hypotheses based on genres, eras, and authors. e.g., the isotopies /countryside/ or /city/, in a text from rural legend. The analytical stage of the analysis selects a few semes or isotopies that are of interest either intrinsically, e.g., the isotopy /aerospace/ in a love story, or because of the relations they maintain with other semes or isotopies. Finally, the comprehensive phases identify the logical relations between isotopies and between semes and identify semic molecules. Here we drop the technical implementations, and we review the semi-automatic implementations. Thus, Cavazza in Rastier-02 makes the semantic analysis in six steps. (0) He classifies the text based on its genre in order to obtain its characteristics. (1) He looks for augmenting the compiled lexies and mentions that a higher inventory has context occurrences. (2) In the second phase, he creates semantic classes (dimensions and domains). However, it is not clear what it is he dubs semantic dimensions. (3) He identifies the minimal semantic classes (semantic fields and taxemes). Here, he compiles the semantic fields by using heuristics (hypothetical-deductive and syntactic cues) and distributional analysis. Once compiled, the semantic fields are divided into taxemes. (4) In this phase, he describes specific and afferent features. He supposes that the description of specific semes is closely linked to the compilation of taxemes. This description can sometimes identify opposed features for quasi-synonymous terms; this makes it possible to describe sememes with features that do not appear very specific. Finally, to verify the description's overall coherence, he compares the descriptions and looks for absence in the contradiction in the same topic.

Tanguy Tanguy-97a develops a remarkable complete work about the IS model. In his implementation of the semi-automatic Human Machine, he develops a process of semantic analysis of three phases: (1) In the first step, the interpreter defines semic units by using its intuition. This way is the semantic traits (semes), the analyzed text chains (spesemes), and the non-restricted associations extracted. (2) Once this is done, the previously defined restrictions are applied to the structure. All the sememes are built at first, associating them with spesemes and the taxemes. In this phase, the restriction of intersection and inclusion of the taxemes are administered. Here, the intervention of an interpreter is also required. (3) In this step, the previously generated and completed taxemes are configured to their oppositions graph organization (specemes). Here, an interpreter's intervention is required to locate a seme and associate it with the relevant speceme. Up to this point, Tanguy determines that a stable structure is set once the defined restrictions are satisfied. The issue with Tanguy's approach is the enormous dependency with the interpreter to generate the semantic units. Nevertheless, the most significant contribution that it makes is the restrictions system (formal model). They will be essential for our model. Later, we will use these restrictions to construct an ontology of the model of IS. Mauceri Mauceri-07 develops an indexer based on the model of IS. His semantic analysis consists of six steps. Just as Rastier-02 constructs a corpus and reads a vocabulary to identify the characteristics of the text elements. Here, different from the previous proposes, it uses co-occurrences around the characteristics of the elements and sets the terminological tags to generate semantic units. Finally, it verifies the consistency of the obtained elements. Phases 3 and 5 are purely automatic. Phases 2, 4, and 6 are exclusively intellectual activities using necessary concordance tools to get back to the text. Other implementations like Coralie's Coralie-08 provide more steps and techniques to the processes previously defined, e.g., Coralie goes more in-depth and speaks about: (1) The corpus cut (or tokenization of the corpus). (2) The assignation of semantic traits based on the formation of morphological groups. (3) To weigh the semantic characteristics. Here, the mathematical treatment given to the analysis is highlighted. Kastberg Kastberg-12 describes a process of extraction of isotopies out of a textual corpus, based on a systematic analysis of semantic structures and co-occurrences, through different textometric programs. They do not improve the model, in any case.

So far, we have reviewed, in the previous works, the development of the semic analysis using different strategies. From a straightforward form through a primary analysis involving only mor-

phemes and their glosses, until more complex, heuristic rules, statistical analysis, extensive corpora, semantic dispersion, or latent semantic mapping methods intervene. However, the semic analysis is an interpretation process that collects and compiles semantic traits and requires adjustments and updates to detect the primary, minimal units of the semantic model. These have been achieved just under the perception of the interpreter. The challenge is how to make a perceptive-of-the-adjustment model, avoiding being bound just to the illusion of the statistics beam. In recent years, we find but a few works of IS, the delay is due to, even though the IS influences a lot, other areas of the model-studies have been proposed, the semiotic, the cognitive semantic, and the compositional semantic. Notwithstanding, the target in Rastier's proposed has always been to have a unique semantic. Thus, we will take the advancements of the previously mentioned areas in favor of the interpretive semantics.

Bibliography

[Altszyler-16] E. Altszyler et al. *Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database*. Tech. rep. Córdoba, Argentina, Sept. 2016, pp. 1–14.

[Arnold-13] P. Arnold. "Semantic Enrichment of Ontology Mappings", in *Advances in Databases and Information Systems*. Genoa, Italy, Sept. 2013, pp. 42–55.

[Baader-10] F. Baader et al., *The description logic handbook: theory, implementation and applications*, 2nd ed. Cambridge UK: Cambridge University Press, 2006, p. 624.

[Bast-15] H. Bast and E. Haussmann. "More accurate question answering on freebase", in. Oct. 2015, pp. 1431–1440.

[Baudis-15] P. Baudiš. "Systems and approaches for question answering", in *Conference and Labs of the Evaluation Forum (CLEF). Toulouse, France*, Sept. 2015, pp. 1–39.

[Berant-14] J. Berant and P. Liang, Semantic parsing via paraphrasing, *In Proceedings of ACL* , vol. 7, pp. 1415–1425, 2014.

[Berners-Lee-01] T. Berners-Lee, J. Hendler, and O. Lassila, The semantic web, *Sci. Am.* , vol. 284.5, p. 2001, May 2001.

[Bertet-17] K. Bertet et al. "14th International Conference (ICFCA)", in *Formal Concept Analysis*. Rennes, France, June 2017, pp. 1–228.

[Biemann-15] C. Biemann and A. Mehler, *Text mining from ontology learning to automated text processing applications*, 1st ed. Darmstadt, Germany: Springer, 2015.

[Bird-09] S Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, 1st ed. New York, USA: O'Reilly, 2009.

[Blackburn-05] P. Blackburn and J. Bos, *Representation and inference for natural language*, 2nd ed. Standford, California: CSLI Publications, 2005.

[Bouslimi-13] Bouslimi R., A. Messaoudi, and A. Akaichi. "Using a bag of words for automatic medical image annotation with a latent semantic," in *Int. J. Artif. Intell. Appl. (IJAIA)*. Dubai, UAE, Nov. 2013, pp. 51–60.

[Brabrand-06] C. Brabrand. "Analyzing ambiguity of context-free grammars," in *International Conference on Implementation and Application of Automata (CIAA)*. Taipei, Taiwan, 2006, pp. 214–225.

[Brachman-04] R. Brachman and H. Levesque, *The Acquisition of Strategic Knowledge*, 1st ed. San Francisco, CA: Morgan Kaufmann, 2004, p. 413.

[Chen-17] D. Chen et al. "Reading Wikipedia to Answer Open-Domain Questions", in *Annual Meeting of the Association for Computational Linguistic*, ed. by Chen-17. Vancouver, Canada, Aug. 2017, pp. 1870–1879.

[Chieu-02] H. Chieu and H. Ng. "Named entity recognition with a maximum entropy approach", in *Proceedings of the 19th international conference on Computational linguistics*. Taipei, Taiwan, Aug. 2002, pp. 160–163.

[Chomsky-57] N. Chomsky, *Syntactic Structures*, 2nd ed. Berlin: Mouton de Gruyter, 1957.

[Cimiano-06] P. Cimiano, *Ontology learning and population from text*, 1st ed. Germany: Springer, 2006, p. 312.

[Cimiano-09] K. Toutanova et al. "Flexible Semantic Composition with DUDES", in *Proceedings of the Eight International Conference on Computational Semantics*. Tilburg, Netherlands, Jan. 2009, pp. 272–276.

[Cohen-16] Shay Cohen, *Bayesian analysis in natural language processing*, 1st ed. Toronto, Canada: Morgan & Claypool, 2016, p. 274.

[Cope-11] B. Cope, M. Kalantzis, and L. Magee, *Towards a semantic web. connecting knowledge in academic research*, 1st ed. Cambridge, UK: Chandos Publishing, 2011, p. 446.

[Coralie-08] R. Coralie. "Analyse et modélisation sémantiques à partir de ressources lexico–sémantiques". PhD dissertation. Ecole des Mines de Paris ATILF, 2008, p. 122.

[Da-Cunha-12] I. Da Cunha et al., DiSeg 1.0: The first system for Spanish discourse segmentation, *Expert Systems with Applications* , vol. 39.2, pp. 1671–1678, Jan. 2012.

[Dayal-16] V. Dayal, *Questions*, 1st ed. Oxford: Oxford University Press, 2016.

[Devlin-18] J. Devlin et al., BERT: pre-training of deep bidirectional transformers for language understanding, *arXiv e-prints* , vol. arXiv:1810, arXiv:1810.04805, 2018.

[Diefenbach-18] D Diefenbach et al., Core techniques of question answering systems over knowledge bases: a survey, *Journal knowledge and information systems* , vol. 55.3, pp. 529–569, 2018.

[Ding-11] Z. Li, Q. Ding, and W. Zhang. "A comparative study of different distances for similarity estimation", in *Intell. Comput. Inf. Sci. (ICICIS)*. Chongqing, China, Jan. 2011, pp. 483–488.

[Dolan-94] J. Völker, P. Hitzler, and P. Cimiano. "Proceedings of the 15th conference on Computational linguistics", in *Word sense ambiguation: clustering related senses*. Kyoto, Japan, Aug. 1994, pp. 712–716.

[Dourish-14] P Dourish, *Ways of Knowing in HCI*, 1st ed. New York, NY: Springer, 2014.

[Dubey-18] M. Dubey and D. Banerjee. "EARL: joint entity and relation linking for question answering over knowledge graphs", in *in International semantic web conference (ISWC 2018)*. Monterrey, CA, 2018, pp. 108–126.

[Elekes-10] A. Elekes, M . Schaeler, and K. Boehm. "On the various semantics of similarity in word embedding models," in *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. TX, USA, June 2017, pp. 1–10.

[Elliott-17] P Elliott et al., Predicates of relevance and theories of question embedding, *Journal of Semantics* , vol. 1, pp. 1–8, Aug. 2017.

[Escobar-18] L. Escobar-Vega, V. Zaldivar-Carrillo, and I. Villalon-Turrubiates. *Comparative analysis and implementation of semantic-based classifier*. Internal Report PhDEngScITESO-18-05-R. Tlaquepaque, Mexico: ITESO, May 2018.

[Euzenat-13] P. Shvaiko and J. Euzenat, *Ontology matching*, 2nd ed. Berlin, Heidelberg: Springer, 2013.

[Faria-11] C. Faria and R. Girardi. "An information extraction process for semi-automatic ontology population", in *Soft Computing Models in Industrial and Environmental Applications, 6th International Conference SOCO*. Salamanca, Spain, Apr. 2011, pp. 319–328.

[Fensel-04] D. Fensel, *Ontologies: a silver bullet for knowledge management and electronic commerce*, 2nd ed. NJ, USA: Springer, 2004, p. 446.

[Fensel-05] D. Fensel et al., *Spinning the semantic web: bringing the world wide web to its full potential*. MA, USA: The MIT Press, 2005.

[Ferdinand-04] M. Ferdinand, C. Zirpins, and D. Trastour. "Lifting XML schema to OWL", in *in International Conference on Web Engineering (ICWE)*, vol. 3140. Munich, Germany, 2004, pp. 354–358.

[Ferrucci-12] D. Ferrucci, This Is Watson, *Journal of Research and Development* , vol. 56, pp. 01–88, 2012.

[Finkel-05] J. Finkel. "Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics", in *Incorporating non-local information into information extraction systems by gibbs sampling*. Ann Arbor, Michigan, June 2005, pp. 363–370.

[Friedman-97] N. Friedman et al., Bayesian network classifiers, *Machine Learning* , vol. 29.2-3, pp. 131–163, Nov. 1997.

[Gabor-16] A. Gábor. "Learning open domain knowledge from text". Ph.D. Thesis. California, US: Dept. of Computer science, Stanford University, Jan. 2016.

[Gamallo-05] P. Gamallo, A. Agustini, and G. Lopes, Clustering Syntactic Positions with Similar Semantic Requirements, *Computational Linguistics* , vol. 31.4, pp. 107–146, May 2005.

[Ganter-05] B. Ganter and G. Stumme, *Formal concept analysis: foundations and applications*, 1st ed. Berlin: Springer, 2005, p. 359.

[Ganter-16] B. Ganter and S. Obiedkov, *Conceptual Exploration*, 1st ed. Berlin: Springer, 2016, p. 359.

[Garcia-05] R. García. "A semantic web approach to digital rights management". Ph.D. Thesis. Barcelona, España: Universitat Pompeu Fabra, Jan. 2005.

[Gardner-18] M. Gardner. "AllenNLP: A Deep Semantic Natural Language Processing Platform", in *Proceedings of Workshop for NLP Open Source Software*. Melbourne, Australia, July 2018, pp. 1–6.

[Geerts-00] G. Geerts and G. William, The ontological foundation of REA enterprise information systems, *Annu. Meet. American Account. Assoc.* , vol. 362.1, pp. 127–150, 2000.

[Goldberg-17] Y. Goldberg, *Neural network methods for natural language processing*, 1st ed. Toronto, Canada: Morgan & Claypool, 2017, p. 287.

[Greimas-66] A. Greimas, *Sémantique structurale. Recherche de méthode*, 1st ed. Paris: Librairie Larousse, 1966.

[Gruber-89] T. Gruber, *The acquisition of strategic knowledge*, 1st ed. London, UK: Academic Press, INC, 1989, p. 337.

[Gruber-95] T. Gruber, Toward principles for the design of ontologies used for knowledge sharing, *International Journal of Human-Computer Studies - Special issue: the role of formal ontology in the information technology* , vol. 43.. 5-6, pp. 907–928, Dec. 1995.

[Grune-08] D. Grune and C. Jacobs, *Parsing techniques*, 2nd ed. Amstelveen, Amsterdam: Springer, 2008.

[Gzyl-95] H. Gzyl, *The Method of Maximum Entropy*, 1st ed. Cambridge UK: Scientific World, 1995, p. 151.

[Hakimov-17] S. Hakimov, S. Jebbara, and P. Cimiano. "Amuse: multilingual semantic parsing for question answering over linked data", in *16th International Semantic Web Conference*. Vienna, Austria, Oct. 2017, pp. 329–346.

[Hall-05] W. Hall, Biological nature of knowledge in the learning organization, *Learn. Organ.* , vol. 12.2, pp. 169–188, Apr. 2005.

[Harispe-15] S. Harispe et al., *Semantic Similarity from Natural Language and Ontology Analysis*, 1st ed. Toronto, Canada: Morgan & Claypool, 2017.

[Harris-54] Z. Harris, Distributional structure, *Word* , vol. 10.2, pp. 146–162, Jan. 1954.

[Hebert-11] L. Hebert, *Tools for text and image analysis: an introduction to applied semiotics*, 1st ed. Rimouski, Quebec: Universite du Quebec a Rimouski, 2011, p. 418.

[Hitzler-11] P. Hitzler, M. Krotzsch, and S. Rudolph, *Foundations of Semantic Web Technologies*, 1st ed. Broken Sound Parkway NW: CRC Press, 2011.

[Hochreiter-97] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation* , vol. 9, pp. 1735–1780, Apr. 1997.

[Horridge-11] M. Horridge and S. Bechhofer, The owlapi: a java api for owl ontologies, *semant. web* , vol. 2, pp. 11–21, Jan. 2011.

[Huang-15] Z. Huang, M. Thint, and Z. Qin. "Question Classification using Head Words and their Hypernyms", in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Oct. 2015, pp. 927–936.

[Huang-18] H. Huang, E. Choi, and W. Yih. "FlowQA: Grasping Flow in History for Conversational Machine Comprehension", in *International Conference on Learning Representations*. New Orleans, US, May 2019, pp. 1–15.

[Ignatov-14] D. Ignatov. "The 8th Russian Summer School in Information Retrieval", in *Introduction to formal concept analysis and its applications in information retrieval and related fields*. Nizhniy Novgorod, Russia, Aug. 2014, pp. 42–141.

[Iyyer-17] M. Iyyer, W. Yih, and M. Chang. "Search-based Neural Structured Learning for Sequential Question Answering," in *Association for Computational Linguistics*. Vancouver, CA, July 2017, pp. 1821–1831.

[Jones-52] W. T. Jones, *A history of western philosophy*, 1st ed. New York, NY: Harcourt, Brace and World, 1952, p. 446.

[Joulin-17] A. Joulin et al. "Fasttext.zip: compressing text classification models", in *5th International Conference on Learning Representations*. Toulon, France, Apr. 2017, pp. 1–13.

[Kallmeyer-10] L. Kallmeyer, *Parsing beyond context-free grammars*, 1st ed. Springer, 2010.

[Kalyanpur-16] A. Kalyanpur, Typing candidate answers using type coercion, *IBM J. Res. Dev*, pp. 1–13, 2016.

[Kamp-85] H. Kamp and U. Reyle, *From discourse to logic*. Dordrecht: Kluwer Academic Publishers, 1985.

[Kastberg-12] M. Kastberg and J. Leblanc, Extraction des isotopies d' un corpus textuel : analyse systématique des structures sémantiques et des cooccurrences , à travers différents logiciels textométriques, *TEXTO!* , vol. 17.3, June 2012.

[Keller-88] W. Keller, Nested cooper storage:the proper treatment of quantification in ordinary noun phrases, *Natural Language Parsing and Linguistic Theories,* , vol. 35, pp. 432–447, Jan. 1988.

[Kusner-15] M. Kusner et al. "From word embeddings to document distances", in *International Conference on Machine Learning*. Lille, France, July 2015, pp. 957–966.

[Kuznetsov-16] V. Kuznetsov and V. Mochalov. "Ontological-semantic text analysis and the question answering system using data from ontology", in *International Conference on Advanced Communication Technology (ICACT)*. Pyeong Chang, Korea, Feb. 2016, pp. 15–18.

[Kwiatkowski-10] T. Kwiatkowski et al. "Inducing probabilistic CCG grammars from logical form with higher-order unification", in *Conference on Empirical Methods in Natural Language Processing, Massachusetts*. Massachusetts, US, Jan. 2010, pp. 1223–1233.

[Lally-16] A. Lally, Question analysis: How Watson reads a clue, *IBM J. Res. Dev.* , vol. 56.3.4, 2:1–2:14, Aug. 2016.

[Landauer-13] T. Landauer, *Handbook of latent semantic analysis*, ed. by Colorado, 1st ed., vol. 104, 2. Psychology Press, Apr. 1997, pp. 211–240.

[Landauer-97] T. Landauer and S. Dumais, A solution to Plato's problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge, *Psychological Review* , vol. 104.2, pp. 211–240, Apr. 1997.

[Lehmann-15] J. Lehmann and J. Völker, DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia, *Semantic Web* , vol. 6, pp. 167–195, Sept. 2015.

[Lenci-08] A. Lenci, Distributional approaches in linguistic and cognitive research, *Italian Journal of Linguistics* , vol. 20.1, pp. 1–31, May 2008.

[Levelt-82] W. Levelt and S. Kelter, Surface form and memory in question answering, *Cogn. Psychol* , vol. 106, pp. 78–106, 1982.

[Levenshtein-66] V. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, *Soviet Physics Doklady* , vol. 10.8, pp. 707–710, Feb. 1966.

[Levrat-16] B. Levrat, B. Kupelioglu, and T. Acarman, Recognition of metonymy by tagging named entities recognition, *WSEAS Transactions on Computer Research* , vol. 4.1, pp. 81–85, Jan. 2016.

[Levy-14] O. Levy and Y. Goldberg. "Dependency-based word embeddings", in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Maryland, USA, June 2014, pp. 302–308.

[Lewis-70] D. Lewis, General semantics, *Synthese* , vol. 22, pp. 18–67, Dec. 1970.

[Lewis-98] D. Lewis. "Naive (bayes) at forty: the independence assumption in information retrieval", in *European Conference on Machine Learning (ECML.* Berlin, Heidelberg, Apr. 1998, pp. 4–15.

[Li-06] L. Xin and D. Roth, Learning question classifiers: the role of semantic information, *Natural Language Engineering* , vol. 12, pp. 229–249, 2006.

[Li-09] J. Li et al., RiMOM: A dynamic multistrategy ontology alignment framework, *IEEE Trans. Knowl. Data Eng.,* , vol. 21.8, pp. 1218–1232, Jan. 2009.

[Liu-98] B. Liu et al. "Integrating classification and association rule mining", in *Knowledge Discovery and Data Mining*. New York, NY, Aug. 1998, pp. 80–86.

[Lloberes-10] M. Lloberes, I. Castellón, and L. Padró. "Spanish Freeling dependency grammar", in *Conference on Language Resources and Evaluation (LREC2010)*. Malta, Italy, May 2010.

[Lohmann-16] S. Lohmann et al., Visualizing ontologies with VOWL, *Semantic Web* , vol. 7.4, pp. 399–419, May 2016.

[Loiseau-18] S. Loiseau. *Textual Data Analysis Package used by the TXM Software*. 2018. URL: `https://cran.r-project.org/web/packages/textometry/textometry.pdf`.

[Lopez-10] V. Lopez et al., Is question answering fit for the semantic web?: A survey, *Semantic Web* , vol. 2, pp. 125–155, 2010.

[Lu-19] X. Lu et al. "Answering complex questions by joining multi-document evidence with quasi knowledge graphs", in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR'19*. Paris, France, July 2019, pp. 105–114.

[Manning-08] C. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*, 1st ed. New York, NY: Cambridge University Press, 2008.

[Mauceri-07] C. Mauceri. "Indexation et isotopie: vers une analyse interprétative des données textuelles". PhD dissertation. L'ecole nationale superieure des telecommnunications de Bretagne, 2007, p. 205.

[Melcuk-12] I. Melcuk, *Semantics: from meaning to text*, 1st ed. Montreal: John Benjamins Pub Co, 2012.

[Mikolov-13a] T. Mikolov et al. "Distributed representations of words and phrases and their compositionality", in *Neural Information Processing Systems (NIPS)*. Nevada, USA, Dec. 2013, pp. 1–9.

[Mikolov-13b] T. Mikolov et al. "Efficient estimation of word representations in vector space", in *Proceedings of the International Conference on Learning Representations*. Scottsdale, AZ, May 2013, pp. 1–12.

[Miller-07] G. Miller and C. Fellbaum, WordNet then and now, *Lang. Resour. Eval* , vol. 41.2, pp. 209–214, Jan. 2007.

[Mohasseb-18] A. Mohasseb, Question categorization and classification using grammar-based approach, *Inf. Process. Manag.* , vol. 54.6, pp. 1228–1243, Aug. 2018.

[Nigam-99] K. Nigam, J. Lafferty, and A. Mccallum. "Using maximum entropy for text classification", in *Machine Learning for Information Filtering (IJCAI),* vol. 1, 1. Stockholm, Sweden, Aug. 1999, pp. 61–67.

[Partee-76] B. Partee, *Montague Grammar*, 1st ed. London: Academic Press, 1976.

[Pasupat-15] P. Pasupat and P. Liang. "Compositional semantic parsing on semi-structured tables", in *International Joint Conference on Natural Language Processing*. Beijing, China, July 2015, pp. 1470–1480.

[Pennington-14] J. Pennington, R. Socher, and C. Manning. "GloVe: global vectors for word representation", in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar, Oct. 2014, pp. 1532–1543.

[Pereira-82] F. Pereira. "Logic for natural language analysis". PhD dissertation. Edinburgh, Scotland: Dept. of Artificial Intelligence, University of Edinburgh, 1982.

[Pincemin-10] B. Pincemin, Semántica interpretativa y textometría, *Semántica e interpretación* , vol. 2.23, pp. 15–55, Jan. 2010.

[Pinker-89] S. Pinker, *Learnability and cognition: the acquisition of argument structure*, 1st ed. London, UK: The MIT Press, 1989.

[Plato-01] Plato, *Phaedo*, 1st ed. Blacksburg, VA: VA: Virginia Tech, 2001, p. 446.

[Popper-72] K. Popper, *Objective knowledge: an evolutionary approach*, 1st ed. New York, NY: Oxford University Press, 1972, p. 390.

[Porter-80] M. Porter, An algorithm for suffix stripping, *Program* 14, pp. 130–137, July 1980.

[Rajpurkar-18] P. Rajpurkar, R. Jia, and P. Liang. "Know what you don't know: Unanswerable questions for SQuAD", in *Association for Computational Linguistics*. Melbourne, Australia, 2018, pp. 784–789.

[Rao-16] J. Rao, H. He, and J. Lin. "Noise-Contrastive Estimation for Answer Selection with Deep Neural Networks", in *International Conference on Information and Knowledge*. Indiana, US, Oct. 2016, pp. 1913–1916.

[Rastier-02] F. Rastier, A. Abeille, and M. Cavazza, *Semantics for Descriptions: from Linguistics to Computer Science*, 1st ed. Chicago, US: CSLI, 2001, p. 359.

[Rastier-04] F. Rastier, Enjeux épistémologiques de la linguistique de corpus, *Corpus Linguistics* , vol. 1.1, pp. 31–45, Jan. 2004.

[Rastier-04a] F. Rastier, Ontologies, Texto!, *Corpus Linguistics* , vol. 18, pp. 15–40, Dec. 2004.

[Rastier-09] F. Rastier, *Sémantique Interprétative*, 3rd ed. Paris FR: PUF, 2009, p. 359.

[Rastier-11] F. Rastier, *La mesure et le grain. Sémantique de Corpus*, 1st ed. Paris FR: Honoré Champio, 2011, p. 359.

[Rastier-96] F. Rastier, *Semántica Interpretativa*, 2nd ed. Paris FR: Siglo XXI editores, 1996, p. 359.

[Reddy-18] S. Reddy, D. Chen, and C. Manning, Coqa: a conversational question answering challenge, *Transactions of the Association for Computational Linguistics* , vol. 7, pp. 1–18, Mar. 2018.

[Russell-10] S. Russell and P. Norvig, *Artificial intelligence a modern approach*, 3rd ed. New Jersey, USA: Pearson, 2010.

[Sadek-14] I. Sadek. "Automatic discrimination of color retinal images using the bag of words approach". Masters Thesis. Le Creusot, France: Dept. Vision and Robotics, Université de Bourgogne, 2014, p. 49.

[Salton-75] G. Salton, A. Wong, and C. Yang, A vector space model for automatic indexing, *Magazine Communications of the ACM* , vol. 18.11, pp. 613–620, Nov. 1975.

[Seaghdha-09] D. Séaghdha. "Semantic classification with WordNet kernels", in *North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT*, vol. 37. Boulder, Colorado, June 2009, pp. 237–240.

[Seo-16] M. Seo et al. "Bidirectional Attention Flow for Machine Comprehension", in *5th International Conference on Learning Representations*, ed. by Seo-16. Toulon, France, Apr. 2016, pp. 1–13.

[Shawe-Taylor-04] J. Shawe-Taylor and C. Nello, *Kernel methods for pattern analysis*, 1st ed. Cambridge UK: Cambridge, 2004, p. 462.

[Shevlyakov-16] G. Shevlyakov, *Robust correlation: theory and applications*, 1st ed. West Sussex, UK: Wiley, 2016, p. 319.

[Shieber-88] S. Shieber, Evidence against the context- freeness of natural language, *Linguist. Philos.* , vol. 8.3, pp. 491–504, 1988.

[Sidorv-14] G. Sidorov et al., Soft similarity and soft cosine measure: similarity of features in vector space model, *Computacion y sistemas* , vol. 18.3, pp. 491–504, 2014.

[Simmons-67] R. Simmons, Answering English questions by computer: a survey, *Communications of the ACM* , vol. 8, pp. 53–70, 1967.

[Spitkovsky-12] V. Spitkovsky and A. Chang. "A cross-lingual dictionary for english wikipedia concepts", in *The International Conference on Language Resources and Evaluation (LREC)*, vol. 22. Jan. 2012, pp. 3168–3175.

[Steedman-01] M. Steedman, *The syntactic process*, 1st ed. Cambridge, Massachusetts: MIT Press, 2001.

[Tang-16] B. Tang, S. Kay, and H. He, Toward optimal feature selection in naive Bayes for text categorization, *International Journal of Artificial Intelligence Applications* , vol. 28.9, pp. 2508–2521, Sept. 2016.

[Tanguy-97a] L. Tanguy. "Traitement automatique de la langue naturelle et interprétation: contribution è l' élaboration d'un modèle informatique de la sèmantique interprétative," PhD Thesis. Rennes, France: Dépt. Intelligence Artificielle et Systèmes Cognitifs,Université de Rennes 1, Rennes, 1997.

[Tanguy-97b] L. Tanguy, Computer-aided language processing : using interpretation to refine man-machine relations, *2nd International Conference on Cognitive Technology* , vol. 8.3–4, pp. 1–11, Aug. 1997.

[Tarski-38] A. Tarski, *Logic, Semantics, Metamathematics Papers From 1923 to 1938*, 2nd ed. London: Oxford: Clarendon Press, 1938.

[Taule-08] M. Taule, A. Martí, and M. Recasens. "Ancora: multilingual and multilevel annotated corpora", in *The international conference on language resources and evaluation*. Marrakech, Morocco, Jan. 2008, pp. 96–101.

[Tesniere-15] L. Tesnière, *Elements of structural syntax*, 1st ed. Amsterdam & Philadelphia: John Benjamins, 2015, p. 446.

[Toutanova-03] K. Toutanova et al. "Feature-rich part-of-speech tagging with a cyclic dependency network", in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. Stroudsburg, PA, June 2003, pp. 173–180.

[Tratz-09] S. Tratz and D. Hovy. "Disambiguation of preposition sense using linguistically motivated features", in *North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT),* Boulder, CO, June 2009, pp. 96–100.

[Tutescu-74] M. Tutescu, *Précis de sémantique française*, 1st ed. Paris FR: Editura Didactica şi Pedagogica, 1974.

[Tversky-78] A. Tversky and G. Itamar. "Studies of similarity". in *Cognition and categorization*. Ed. by E Rosch and B Lloyd. 1st ed. Hillsdale, NJ: Lawrence Elbaum Associates, 1978. Chap. 4, pp. 79–98.

[Unger-11] C. Unger and P. Cimiano, Pythia: compositional meaning construction for ontology-based question answering on the semantic web, *Natural Language Processing and Information* , vol. 4.6716, pp. 153–160, June 2011.

[Uwe-88] R. Uwe and C. Rohrer, *Natural language parsing and linguistic theories*, 1st ed. Boston, US.: Springer, 1988.

[Völker-07] J. Völker, P. Hitzler, and P. Cimiano. "Acquisition of OWL DL axioms from lexical resources", in *4th European Semantic Web Conference*. Innsbruck, Austria, June 2007, pp. 670–685.

[Wang-16] Z. Wang et al., Multi-Perspective Context Matching for Machine Comprehension, *arXiv e-prints, p. arXiv:1612.04211*, 2016.

[Wille-82] R. Wille. "Restructuring lattice theory: an approach based on hierarchies of concepts", in *Ordered Sets*. Canada: Springer, Sept. 1982, pp. 445–470.

[Williams-17] J. Williams and G. Santia. "Context-sensitive recognition for emerging and rare entities", in *The 3rd Workshop on Noisy User-generated Text (W-NUT)*. Copenhagen, Denmark, Sept. 2017, pp. 172–176.

[Xu-14] Y. Xu and B. Van Durme. "Information extraction over structured data: question answering with freebase", in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, vol. 1. Baltimore, Maryland, June 2014, pp. 956–966.

[Zhang-14] Quan Zhang et al., Semantic conceptual primitives computing in text classification, *Proceedings of the International Conference on Asian Language Processing 2014, IALP 2014* , vol. 1.1, pp. 215–218, Oct. 2014.

Index of Authors

Subject Index