Detección y extracción de neologismos semánticos especializados

Un acercamiento mediante clasificación automática de documentos y estrategias de aprendizaje profundo

Andrés Torres Rivera

TESI DOCTORAL UPF / ANY 2019

Directors de la tesi
Dra. Rosa Estopà Bagot
Dr. Juan-Manuel Torres-Moreno
Departament de Traducció i Ciències del Llenguatge
LIA – Laboratoire Informatique d'Avignon

A Ana y Roberto

Agradecimientos

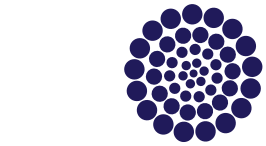
En primer lugar quiero expresar mi gratitud a los directores de esta tesis doctoral, la Dra. Rosa Estopà Bagot y el Dr. Juan-Manuel Torres-Moreno, quienes además de aportarme las herramientas para sacar adelante este proyecto, mostraron un gran carácter humano durante los momentos más duros de este proceso. Más allá de lo académico, me alegro enormemente de haber podido contar con ambos.

Al Institut Universitari de Lingüística Aplicada por darme la oportunidad de formar parte de este gran equipo. También al Observatori de Neologia donde encontré un sitio de trabajo e igualmente al Laboratoire d'Informatique d'Avignon cuya infraestructura fue indispensable. Desde luego a todos miembros del IULA, Teresa Cabré, Mercè Lorente, Judit Freixa, Ona Domènech, Nuria Bel y Amor Montané. Y en particular a Jorge Vivaldi e Iria da Cunha por el interés que mostraron en este proyecto y por todos los consejos y recomendaciones. Igualmente a Everardo Mendoza y Maritza López, mis profesores de la Universidad Autónoma de Sinaloa, por introducirme en el mundo de la lexicografía y motivarme a estudiar un posgrado.

Muy especialmente a Marina, por estar a mi lado en las buenas y en las malas, en la salud y en la enfermedad, hoy y siempre. Por ser pilar cuando he flaqueado y luz en los momentos más oscuros. Agradezco tu apoyo, paciencia y comprensión, pero sobre todas las cosas, tu compañía y cariño, que me han dado la fuerza para seguir adelante. A mi madre —mi familia— por creer en mí y confiar en que todo saldría bien, me has animado y apoyado desde lejos, siempre y a pesar de todo. Una vez más tuviste razón, las cosas encontraron su orden natural a pesar de no haber seguido el camino ideal. A Paquita, Ester, Yolanda y Óscar, quienes me abrieron las puertas de un hogar y me han dado todo su apoyo. Ya saben que no tengo palabras suficientes para expresar la gratitud que tengo por todo lo que he recibido.

A Xurxo Diz quem sempre tivo unha perspectiva innovadora e creativa e tomou o tempo para ver o meu traballo. Non sei quen é o narrador desta aventura, pero espero que nos permita coincidir de novo. A Alba, Xevi, Laura i Carlos, per brindar-me la seva amistat, mai han dubtat a tirar-me una mà (o una crossa), els dec moltes. Als meus companys de l'OBNEO i INFOLEX, Ivan, Martí, Pedro, Eli i Adriana, omplen aquestes despatxs de coneixement i companyerisme. Molts ànims i mantinguin l'esperit, saben que poden comptar amb mi per debatre qüestions lingüístiques, tot i que l'Ivan i jo tindrem la raó, com sempre. También a todas las compañeras de Gr@el, Marta, Yris, Aina, Ana, Lucy, Patricia y Emmy, siempre han tenido el consejo preciso y han sido un gran grupo de apoyo, aprecio mucho que me hayan brindado amistad.

A mis amigos de México en Barcelona, Paco, Blanca, Dani, Jorge y Juanita. Compartimos piso, universidad, aventuras, angustias y alegrías, que no se les olvide que tenemos unos tequilas que dejamos en "ahorita". À Paty, Pamela, Sabine, Julien et mes compagnons du LIA, Carlos, Adrián, Luis, Elvys, Nacho et Mathias, pour le support, les moments et rencontres intra (et extra) murs que nous avons passés en Avignon. Mecs, n'oubliez pas que des samedis le LIA est également ouvert, mais vous devez l'ouvrir avec précaution pour que l'alarme ne soit pas activée, ce qui n'est jamais arrivé à personne.



Esta tesis se ha desarrollado en el marco del programa de Becas al Extranjero convocatoria 2015 del Consejo Nacional de Ciencia y Tecnología (CONACYT) del gobierno de México.

Resumen

En el campo de la neología, se han desarrollado diferentes acercamientos metodológicos para la detección y extracción de neologismos semánticos empleando estrategias como la desambiguación semántica y el modelado de temas, pero todavía no existe una propuesta de un sistema para la detección de estas unidades. A partir de un estudio detallado sobre los supuestos teóricos necesarios para delimitar y describir los neologismos semánticos, en esta tesis proponemos el desarrollo de una aplicación para identificar y vaciar dichas unidades mediante estrategias estadísticas, de minería de datos y de aprendizaje automático. La metodología planteada se basa en el tratamiento del proceso de detección y extracción como un problema de clasificación, que consiste en analizar la concordancia de temas entre el campo semántico del significado principal de una palabra y el texto en el que se encuentra. Para constituir la arquitectura del sistema propuesto, analizamos cinco métodos de clasificación automática supervisada y tres modelos para la generación de representaciones vectoriales de palabras mediante aprendizaje profundo. Nuestro corpus de análisis está compuesto por los neologismos semánticos del ámbito de la informática pertenecientes a la base datos del Observatorio de Neologia de la Universitat Pompeu Fabra, que han sido registrados desde 1989 hasta 2015. Utilizamos este corpus para evaluar los distintos métodos que implementa el sistema: clasificación automática, extracción de palabras a partir de contextos cortos y generación de listas de palabras similares. Este primer acercamiento metodológico busca establecer un marco de referencia en materia de detección y extracción de neologismos semánticos.

Resum

Dins del camp de la neologia, s'han dissenyat diferents aproximacions metodològics per a la detecció i extracció de neologismes semàntics amb tècniques com la desambiguació semàntica i el modelatge de temes, però encara no existeix cap proposta d'un sistema per a la detecció d'aquestes unitats. A partir d'un estudi detallat sobre els supòsits teòrics necessaris per identificar i descriure els neologismes semàntics, en aquesta tesi proposem el desenvolupament d'una aplicació per identificar i buidar aquestes unitats mitjançant estratègies estadístiques, de mineria de dades i d'aprenentatge automàtic. La metodologia que es planteja es basa en el tractament del procés de detecció i extracció com un problema de classificació, que consisteix a analitzar la concordança de temes entre el camp semàntic del significat principal d'una paraula i el text en què es troba aquesta paraula. Per constituir l'arquitectura del sistema proposat, analitzem cinc mètodes de classificació automàtica supervisada i tres models per a la generació de representacions vectorials de paraules mitjançant aprenentatge profund. El nostre corpus d'anàlisi està format pels neologismes semàntics de l'àmbit de la informàtica pertanyents a la base de dades de l'Observatori de Neologia de la Universitat Pompeu Fabra, que s'han registrat des de 1989 fins a 2015. Utilitzem aquest corpus per avaluar els diferents mètodes que implementa el sistema: classificació automàtica, extracció de paraules a partir de contextos breus i generació de llistes de paraules similars. Aquesta primera aproximació metodològica busca establir un marc de referència en matèria de detecció i extracció de neologismes semàntics.

Résumé

Dans le domaine de la néologie, différentes approches méthodologiques ont été développées pour la détection et l'extraction de néologismes sémantiques. Ces approches utilisent des stratégies telles que la désambiguïsation sémantique et la modélisation thématique, mais il n'existe aucun système complet de détection de néologismes sémantiques. Ainsi, nous proposons dans cette thèse le développement des algorithmes qui permettent d'identifier et d'extraire les néologismes sémantiques au moyen de méthodes statistiques, d'extraction d'information et d'apprentissage automatique. La méthodologie proposée est basée sur le traitement du processus de détection et d'extraction en tant que problème de classification. Il consiste à analyser la proximité des thèmes entre le champ sémantique de la signification principale d'un terme et son contexte. Pour la construction du système nous avons étudié cinq méthodes de classification automatique supervisée et trois modèles pour la génération de représentations vectorielles de mots par apprentissage profonde. Le corpus d'analyse est composé de néologismes sémantiques du domaine informatique appartenant à la base de données de l'Observatoire de Néologie de l'Université Pompeu Fabra, enregistrés de 1989 à 2015. Nous utilisons ce corpus pour évaluer les différentes méthodes mises en œuvre par le système : classification automatique, extraction de mots à partir de contextes courts et génération de listes de mots similaires. Cette première approche méthodologique cherche à établir un cadre de référence en termes de détection et d'extraction de néologismes sémantiques.

Abstract

In the field of neology, different methodological approaches for the detection and extraction of semantic neologisms have been developed using strategies such as word sense disambiguation and topic modeling, but there is still not a proposal for a system for the detection of these units. Beginning from a detailed study on the necessary theoretical assumptions required to delimit and describe semantic neologisms, in this thesis, we propose the development of an application to identify and extract said units using statistical, data mining and machine learning strategies. The proposed methodology is based on treating the process of detection and extraction as a classification task, which consists on analyzing the concordance of topics between the semantic field from the main meaning of a word and the text where it is found. To build the architecture of the proposed system, we analyzed five automatic classification methods and three deep learning based word embedding models. Our analysis corpus is composed of the semantic neologisms of the computer science field belonging to the database of the Observatory of Neology of the Pompeu Fabra University, which have been registered from 1989 to 2015. We used this corpus to evaluate the different methods that our system implements: automatic classification, keyword extraction from short contexts, and similarity list generation. This first methodological approach aims to establish a framework of reference in terms of detection and extraction of semantic neologisms.

Índice general

Índice de figuras

Índice de tablas

Lista de ecuaciones

Capítulo 1

INTRODUCCIÓN

En la actualidad, el uso de aplicaciones informáticas para detectar y extraer neologismos es fundamental. Observatorios de neología como el OBNEO[1], NEOPORTERM[2] y OBNEQ[3] emplean herramientas informáticas para llevar a cabo las tareas de detección y extracción de forma más efectiva y, así, mejorar los resultados obtenidos. Estas aplicaciones están especializadas principalmente en la detección de neologismos formales (NF), dado que la detección de forma automática o semiautomática de neologismos semánticos (NS) es más complicada debido a sus características lingüísticas.

Hasta 2015, año en el que se presenta Logoscope[4], solamente existía otro sistema especializado en la detección de NS, April[5]. Ambas herramientas aplican dos estrategias diferentes para realizar la detección de NS. April funciona con reglas estadísticas y patrones de colocación analizados cronológicamente, mientras que Logoscope emplea la detección automática de temas para determinar las palabras asociadas a cada temática, así, cuando una palabra es detectada en una temática diferente a las regulares o prototípicas, el sistema la señala como candidata a NS. Asimismo, existen propuestas metodológicas que han abordado la tarea de detección de NS, como los estudios de Janssen (2009) o de Nazar (2010). Ambas aproximaciones, a pesar de no tener como resultado una aplicación destinada a la detección de NS, son interesantes desde el punto de vista metodológico, ya que proponen acercamientos distintos a los empleados por April y Logoscope. Entre los puntos en común que tienen estos sistemas y propuestas metodológicas, se pueden mencionar el uso de métodos estadísticos, de reglas heurísticas, de minería de texto y de aprendizaje automático.

No obstante, dichos sistemas y acercamientos metodológicos no terminan de ser, en general, eficaces y, en consecuencia, la metodología de trabajo para detectar NS que se sigue en el OBNEO todavía es manual y consiste en analizar las fuentes de prensa para registrar la aparición de candidatos. Una vez registrados, se revisan sus acepciones en los diccionarios de referencia para comparar los significados lexicográficos de cada acepción con el posible NS. En el caso de que dicha palabra sea un candidato válido, se indica con respecto a qué acepción o acepciones se considera NS (Cabré y Estopà, 2004b).

[1] Observatori de Neologia (OBNEO) `https://www.upf.edu/web/obneo/`

[2] Observatório de Neologia e de Terminologia em Língua Portuguesa (NEOPORTERM).

[3] Observatoire de Néologie du Québec (OBNEQ) `http://www.lli.ulaval.ca/recherche/groupes-et-laboratoires`

[4] `http://logoscope.unistra.fr`

[5] `http://rdues.bcu.ac.uk/aprdemo/`

En nuestra investigación presentamos un sistema (denominado DENISE) que combina las cuatro estrategias —mencionadas anteriormente— para automatizar el proceso de detección y clasificación de los NS. DENISE es un sistema capaz de determinar si una concordancia contiene un término candidato a NS analizando, por una parte, su temática especializada mediante un modelo de clasificación automática de documentos y evaluando, por otra parte, su temática general por medio de un campo semántico generado gracias a la similitud de palabras en un modelo neuronal de lengua. La concordancia de temas entre los dos modelos indica que la unidad léxica en cuestión no es un candidato válido, mientras que la discordancia puede significar que dicha unidad sí es un candidato a NS.

Esta tesis tiene una orientación teórica y aplicada, dado que presentamos un sistema que es resultado de una propuesta metodológica, que, a su vez, ha sido diseñada a partir de la revisión teórica. La orientación teórica es indispensable para comprender qué es un NS y cuáles son los mecanismos lingüísticos que lo identifican en el discurso. Por su parte, la aplicación de algoritmos de aprendizaje automático constituye la base sobre la que se sustenta DENISE, ya que dichos algoritmos realizan un análisis de tema y significado para detectar y extraer los candidatos a NS.

1.1 Antecedentes

En relación con las metodologías de extracción de neologismos, en el trabajo de fin de máster (Torres, 2015) (en adelante TFM) ya se constató que el criterio lexicográfico es el empleado con mayor frecuencia. Este criterio consiste en detectar la presencia de una unidad en un corpus de exclusión compuesto por lemarios representativos de una lengua dada (Vivaldi, 2003; Cabré, 2004; Domènech, 2008). Sin embargo, en el trabajo de Nazar (2010), se observa que este criterio tiene limitaciones frente a los siguientes tipos de unidades o candidatos:

- Candidatos a neologismo que no coinciden con la idea que tienen los hablantes sobre esta unidad nueva.
- Unidades de estructura sintagmática que no corresponden a una entrada propia en los diccionarios.
- Los neologismos semánticos, dado que estos aparecen registrados en el corpus de exclusión, pero con otro sentido.
- Unidades obtenidas de vaciado manual, ya que el informante puede no conocer una unidad que existe desde hace tiempo, puesto que esta se encuentra fuera de su área de dominio, ya sea por motivos generacionales o culturales.

Para dar solución a estos problemas, los enfoques modernos proponen estrategias computacionales y estadísticas. En el estudio de Nazar encontramos tres líneas de investigación basadas en métodos estadísticos, cuya finalidad es presentar una metodología de análisis cuantitativo que permita estudiar la neología empleando técnicas informáticas para automatizar este proceso:

- La primera línea propone el uso de filtros que permitan establecer grados de neologicidad, de forma que haya una aproximación a lo que se percibe como novedad en la lengua mediante un estudio diacrónico de distribución de frecuencias.

- La segunda línea corresponde a la detección de combinaciones de palabras.

- La tercera línea, enfocada en el caso de la NS, consiste en realizar un estudio de concurrencias, es decir, de la aparición conjunta de dos unidades a una distancia flexible, sin importar el orden dentro de una ventana de contexto de *n* (una cantidad a determinar) palabras.

Por su parte, Janssen (2009) propone criterios para la detección semiautomática de candidatos a neologismos. Define *candidatos* como palabras que probablemente son neologismos, pero para determinarlo con certeza aún es necesaria la intervención humana. Estos criterios se consideran semiautomáticos, ya que para ser considerados automáticos, tendrían que incorporar procesos de filtrado con el fin de automatizar las decisiones y la clasificación. Los criterios para la detección de neologismos propuestos por Janssen son los siguientes:

- Utilizar una lista independiente de palabras conocidas: mantener una lista de palabras conocidas (lista de exclusión). Los candidatos a neologismos serán aquellas palabras que se encuentren dentro de corpus de referencia y que no estén documentadas en la lista de exclusión.

- Utilizar patrones lingüísticos que caracterizan a los neologismos: un neologismo puede ser reconocido al analizar las palabras de su contexto. Esta idea se basa en que los neologismos son palabras poco conocidas para los lectores y para facilitar su comprensión se introducen de forma especial (por ejemplo, mediante marcadores discursivos) en los textos.

- Contar las ocurrencias de las palabras del corpus de estudio: realizar el conteo de frecuencias en comparación con el corpus de referencia.

El último punto de los criterios propuestos por Janssen se desglosa en cuatro subapartados que describen las principales metodologías estadísticas empleadas:

- La primera está basada en el análisis de *hapax legomena* implementado por Renouf (1998), es decir, son neologismos las palabras que aparecen solamente una vez en un corpus de referencia. Este método utiliza concordancias de palabras y no es propiamente un abordaje estadístico.

- La segunda medida estadística es la frecuencia cero, todos aquellas palabras que aparezcan en un corpus de estudio y no aparezcan en un corpus de referencia son candidatos a neologismo.

- El tercer método es la frecuencia de todas las palabras en todos los corpus que se están utilizando. Cuanto mayor sea la frecuencia de una palabra en el corpus de estudio, en comparación con su frecuencia en el corpus de referencia, es probable que se trate de un neologismo. Una variante a este método es la mencionada en

Nazar (2010), que consiste en dividir los corpus de referencia diacrónicamente para graficar el incremento de su uso. En este caso las palabras que tengan un mayor crecimiento serán los candidatos a NS.

- La cuarta estrategia es contar la frecuencia de las palabras que ocurren en un contexto determinado. Cuando el contexto convencional de una palabra se modifica (aparece con palabras con las que no suele colocarse), este cambio es indicio de un posible nuevo significado de dicha palabra.

1.2 Ideas previas

Después de realizar un análisis de las herramientas y métodos para detectar neologismos en el TFM, se pueden considerar las siguientes ideas como bases teóricas para el desarrollo de un sistema cuya finalidad sea la detección de NS:

- La detección automática de NS, por su naturaleza, es más complicada en comparación con otros tipos de neologismos, como los neologismos formales (Tebé, 2002; Sablayrolles, 2006; Janssen, 2009; Renouf, 2010; Reutenauer et al., 2011). En consecuencia, en la actualidad existe una necesidad cubierta de forma parcial.

- La combinación de estrategias lingüísticas, estadísticas y de minería de texto y datos han dado buenos resultados en la detección y extracción de otros tipos de neologismos, principalmente neologismos formales (Pecina, 2009).

- En otras áreas de la lingüística como la terminología (Kobayashi y Takeda, 2000; Cabré et al., 2001; Jurafsky y Martin, 2009), las estrategias de aprendizaje automático y redes neuronales se utilizan para resolver problemas de clasificación de documentos, extracción de información y generación de resúmenes automáticos (Baeza-Yates y Ribeiro-Neto, 1999; Torres-Moreno et al., 2001, 2002; Molina et al., 2010; Torres-Moreno, 2011, 2014; Gérard et al., 2014).

- La desambiguación semántica y de significado tiene un rol importante en el desarrollo de la aplicaciones que tratan con cambio semántico, polisemia o metáforas (Patwardhan et al., 2003; Legrand et al., 2003; Pecina, 2009). En este caso, la desambiguación ayuda a determinar con certeza la existencia una unidad neológica en un texto.

- Los modelos de espacios vectoriales (Salton et al., 1975; Dubin, 2004; Manwar et al., 2012; Xu et al., 2016) para el análisis de relevancia de información entre documentos, puede implementarse para analizar el contenido semántico y similitud entre textos.

1.3 Objeto de estudio

Limitamos el alcance de esta tesis a neologismos semánticos que provienen del campo especializado de la informática, sin especificar el tipo de mecanismo que produce la resemantización. En Cabré (2011b) se muestra que el proceso de cambio semántico puede

responder a diferentes mecanismos. Esta clasificación asigna para cada tipo neologismo un perfil general, en el que cada criterio del perfil cuenta con un valor marcado o no marcado. El apartado que corresponde a los NS se subdivide en tres posibles mecanismos: reducción de significado, ampliación de significado y cambio semántico. Siguiendo esta propuesta, limitamos el análisis hasta el nivel de resemantización (ver tabla 1.1) sin analizar el proceso particular del cambio, ya que la finalidad de la aplicación será detectar un nuevo uso semántico y no precisar si corresponde a una ampliación de significado o a una reducción de significado. En el capítulo 3 se analizará a profundidad el concepto de *neologismo semántico*, cómo se ha clasificado a lo largo del tiempo y por qué nos apegamos a esta definición.

Formación	Cambio	Resemantización	Reducción de significado
			Ampliación de significado
			Cambio de significado

Tabla 1.1 – Fragmento de la tabla de clasificación multivariante (Cabré, 2011b, p. 485).

1.4 Objetivos de la tesis

El objetivo general de la tesis es proponer una metodología de extracción y detección de NS que provienen del ámbito de la informática y que se integran en la lengua general con un significado neológico. Para analizar la inclusión de estas unidades en la lengua general, utilizamos una combinación estrategias de aprendizaje automático supervisado y no supervisado, en específico: modelos de clasificación automática, algoritmos para la extracción de palabras y representaciones vectoriales de palabras. Además, consideramos los siguientes objetivos específicos:

- Analizar en profundidad las características que identifican los NS.
- Investigar las aproximaciones actuales para la detección y extracción de NS.
- Diseñar un algoritmo que permita la detección y extracción de NS.
- Evaluar el funcionamiento de dicho algoritmo.
- Desarrollar una herramienta que implemente dicho algoritmo y realice la detección y extracción semiautomática de NS.

1.5 El sistema DENISE

DENISE será una aplicación que implemente una aproximación estadística complementada con estrategias lingüísticas, de minería de texto y datos, y aprendizaje automático. Empleará corpus generales, especializados, listas de términos y campos semánticos como materiales de trabajo. Con estos elementos, DENISE tratará la detección de neologismos semánticos como un problema de clasificación. Para este fin, el sistema propuesto contará con las siguientes características:

- Multilingüe: Las lenguas de trabajo iniciales son catalán, español y francés, pero podrá ser implementado con facilidad en cualquier otra lengua mientras se cuente con todos los elementos de trabajo necesarios.

- Modular: Su estructura permitirá una fácil adición de nuevas lenguas de trabajo, algoritmos de clasificación y similitud, metodologías para la extracción de palabras u otras funcionalidades como el trabajo colaborativo.

- Multiplataforma: Los lenguajes de programación en los que de desarrollará permitirán la creación de una interfaz de usuario a través de una implementación web.

1.6 Interés y aplicación de la tesis

El interés principal de este trabajo es, por una parte, el desarrollo de una metodología para la detección de NS en corpus de prensa y, por otra, el desarrollo de una aplicación que implemente esta metodología para facilitar el trabajo de los observatorios de neología. Esta también sería una aplicación inmediata, dado que el sistema DENISE podría complementar la plataforma OBNEO para así crear un sistema más robusto.

Para generar los campos semánticos se entrenaron tres diferentes modelos neuronales de lengua. Mediante estas representaciones vectoriales, podemos calcular la similitud que cada palabra tiene con el resto del vocabulario y, así, obtener listados con las unidades más similares al espacio vectorial de la palabra consultada. Estos campos podrían servir también para la identificación de temáticas en textos, currículos, perfiles laborales, etc.

Contar con un sistema multilingüe permite realizar estudios comparativos y el trabajo colaborativo. Al detectar un NS en una lengua A, se puede analizar si la resemantización ha ocurrido paralelamente en una lengua B, o viceversa. Estos estudios también pueden servir como indicador de la implantación de unidades terminológicas en la lengua general.

Finalmente, desarrollar un sistema que detecte NS sería un primer paso hacia la detección de figuras retóricas como la metáfora, la metonimia u otros recursos lingüísticos que dependan de la desambiguación semántica y temática para determinar el significado de una unidad determinada.

1.7 Estructura de la tesis

La presente tesis se encuentra distribuida en nueve capítulos, incluyendo este primer capítulo de introducción. La finalidad de este primer capítulo ha consistido en presentar, de forma general, los antecedentes teóricos y aplicados que fundamentan la presente tesis, de forma que el lector pueda situarse en el contexto de trabajo en que se desarrolla el proyecto. El capítulo 2 presenta el estado de la cuestión en materia de detección de neologismos semánticos, en este capítulo analizamos las propuestas contemporáneas para comprender las metodologías, métodos de evaluación y resultados de cada enfoque. Este apartado, además de mostrar un panorama general de la materia, sirve como punto de referencia para nuestra propuesta metodológica, ya que el diseño de nuestra aplicación parte de supuestos metodológicos comunes.

En el capítulo 3 (marco teórico) presentamos una cronología sobre el concepto de neologismo semántico desde diversas escuelas de pensamiento. El propósito de esta cronología es analizar los puntos comunes entre cada posición, así como los mecanismos de creación de neologismos semánticos. En segundo término mostramos dos corrientes teóricas contemporáneas sobre la definición y clasificación de los neologismos semánticos. Este contraste nos ha permitido justificar la selección de nuestro enfoque teórico. Finalmente, dado que esta tesis se acota a neologismos semánticos terminológicos, analizamos los conceptos claves de la teoría comunicativa de la terminología: la definición de unidad terminológica y el principio de adecuación.

En la metodología (capítulo 4) describimos nuestro enfoque de trabajo, que consiste en el uso de técnicas de clasificación de tema basadas, principalmente, en modelos de aprendizaje automático supervisado, algoritmos de extracción de palabras basados en grafos y el uso de representaciones vectoriales de palabras. En este apartado también presentamos un breve resumen sobre conceptos de aprendizaje automático, así como los componentes básicos de nuestro sistema en conjunto con la descripción del flujo de trabajo de la aplicación.

Posteriormente, en la descripción del sistema (capítulo 5) presentamos un acercamiento preliminar que ha servido como línea base y, posteriormente, analizamos en profundidad los componentes que se requieren para el desarrollo de nuestra aplicación final. Describimos los recursos que han sido necesarios para el entrenamiento y evaluación de los diferentes modelos implementados en cada etapa de análisis de nuestro sistema y, en los capítulos siguientes, describimos los algoritmos y métodos que fueron evaluados para justificar la selección de nuestra metodología definitiva.

A continuación, en el capítulo 6 llevamos a cabo experimentos de clasificación y extracción de palabras a partir de contextos que contienen neologismos semánticos previamente detectados. Evaluamos la efectividad de las implementaciones para detección de lengua (Langdetect) de trabajo y extracción de palabras (TextRank) empleando dichos contextos. Por otra parte, en el apartado relativo a la detección de temas, comparamos cinco arquitecturas de modelos de clasificación para seleccionar un modelo basado en la función de regresión logística.

Dentro del mismo capítulo, evaluamos tres modelos diferentes para la generación de *embeddings*: Word2Vec, FastText y Sense2Vec. Con cada modelo generamos listados de palabras similares, para emplearlos como campos semánticos que dan cuenta de la temática principal y el significado básico de una palabra. Analizamos la efectividad de esta metodología como un problema de clasificación binaria, de forma que a cada campo semántico se debería asignar la temática correspondiente.

En el capítulo 7 describimos la implementación web de nuestro sistema, describimos la interacción con el usuario, el tipo de datos de entrada que emplea y el reporte de resultados que genera. En conjunto con esta descripción, mostramos casos de uso en cada lengua de trabajo utilizando textos obtenidos de fuentes de prensa. Estos casos de uso sirven para ilustrar la operación del sistema, el proceso de detección de neologismos semánticos y la selección final de candidatos.

Finalmente, en las conclusiones (capítulos 8 y 9, en francés) resumimos los resultados obtenidos durante el análisis y comprobamos el cumplimiento de los objetivos de la tesis. También analizamos las limitaciones, posibles mejoras de nuestro enfoque y las líneas futuras investigación que pueden ser de interés.

Capítulo 2

ESTADO DE LA CUESTIÓN

Los siguientes subapartados presentan de forma general los acercamientos actuales en materia de detección y extracción de NS. Los dos primeros presentan herramientas terminadas: April, tiene como finalidad específica detectar NS usando un corpus de prensa y patrones de colocación; y el segundo, Logoscope, emplea modelos de tópicos y una metodología de clasificación que puede ser usada para la detección de NS. También existen otros acercamientos que abordan el tratamiento semiautomático para la detección de NS, por ejemplo la metodología propuesta por Maarten Janssen que consiste en la complementación de las reglas estadísticas empleadas por un sistema etiquetador, que tiene como subproducto la detección de NS. También se pueden destacar Rogelio Nazar, cuyos trabajos presentan ideas y metodologías que emplean inducción de significado (WSI) y *clustering* que podrían ser implementadas para diseñar una herramienta para detectar neologismos semánticos (NS).

2.1 April

Aviator[1] (1990-1993) y April[2] (1997-2000) son dos proyectos realizados por el Research & Development Unit for English Studies de la Universidad de Birmingham City[3]. Aviator se enfoca en la extracción de la información que se planea analizar y April es un sistema de detección de neologismos a nivel lexicológico y semántico. Ambos sistemas emplean corpus de prensa en inglés. A diferencia de otros sistemas April no depende de reglas gramaticales o formales, sino que el análisis se lleva a cabo mediante cuatro filtros que permiten detectar los cambios de significado (Renouf, 2010).

Los filtros se basan en la hipótesis de que los patrones superficiales del texto, en particular los patrones de colocación de una palabra, determinan el sentido de la unidad o palabra en cuestión. Por lo tanto, las diferencias en el contexto de aparición de una palabra a través del tiempo determinan el cambio de significado en las palabras existentes. Así, bajo este criterio, se determina la NS y el cambio semántico de palabras ya existentes (Renouf, 2012). La aplicación se basa en análisis de patrones sintácticos para extraer neologismos de forma automática. Estos pueden ser palabras nuevas, sentidos nuevos

[1] *Analysis of Verbal Interaction and Automated Text Retrieval*.
[2] *Analysis and Prediction of Innovation in the Lexicon*.
[3] `http://rdues.bcu.ac.uk/`

de una palabra existente, palabras compuestas o sinónimos nuevos (Renouf, 2010). La secuencia de los filtros se nuestra en la figura 2.1.

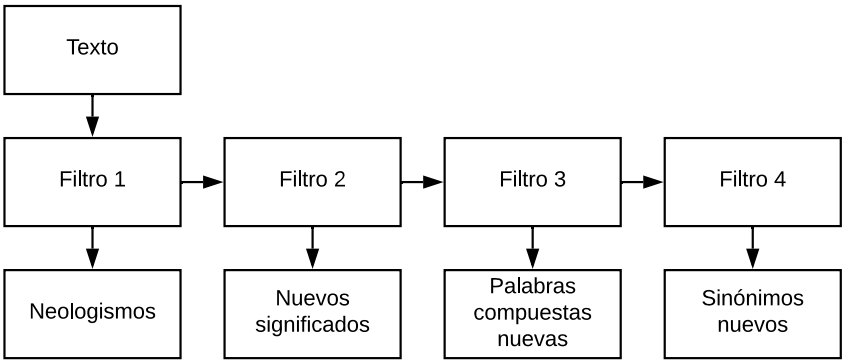


Figura 2.1 – Secuencia de filtros de April (Renouf, 2010, p. 129).

El primer filtro se encarga de la identificación y extracción de neologismos. Para realizar esta tarea, la entrada de datos consiste en corpus trimestrales que se comparan contra un corpus de datos de fecha precedente al del corpus que es ingresado y lo realiza de tres maneras diferentes:

- Por *bootstrapping*, es decir, por la comparación de las palabras de los textos del corpus trimestral ingresado, los del último periodo contra los del periodo anterior, con la finalidad de encontrar las diferencias léxicas entre ambos corpus. De esta forma, los resultados a través del tiempo se acotan.
- Por construcción de una base de control formal de los datos precedentes, es decir, el corpus cronológicamente anterior al periodo estudiado con el cual se comparan los datos del corpus ingresado para extraer las diferencias entre ambos.
- Por utilización de un diccionario de referencia o combinación de un diccionario y un léxico extraído de los primeros textos del corpus. Con este corpus se comparan las palabras contenidas en los textos del periodo que está siendo analizado con el fin de extraer las diferencias.

Tras realizar esos tres procesos se registran los neologismos. Cada candidato es registrado trimestralmente para observar la evolución de los candidatos a través del tiempo ya que de esta forma se lleva control del análisis de colocaciones. En una matriz de dos ejes, el eje horizontal corresponde al perfil de aparición y frecuencia de cada palabra a través del tiempo; y el eje vertical corresponde a la distribución sincrónica del léxico. A todas estas palabras se les atribuyen categorías léxicas preliminares: palabras comunes, nombres propios, abreviaturas, palabras ambiguas y cifras.

Posteriormente se determinan las categorías gramaticales, cada palabra se compara con un diccionario y los afijos conocidos se actualizan constantemente. Finalmente, para determinar los afijos, se revisan cada uno de los caracteres para detectar las raíces correspondientes y filtrar las nuevas combinaciones.

El segundo filtro consiste en la identificación y extracción de NS mediante un análisis de colocaciones de todas las palabras del corpus tomando en cuenta cuatro unidades a ambos extremos del nodo, por ejemplo:

Mary had a little lamb, its fleece was white.[4]

Para llevar a cabo el registro de las colocaciones se propone el uso de una *ventana móvil*. Esto consiste en agrupar los patrones de colocación de cada *nodo* con —entendiendo como *nodo* una palabra— su contexto en una sola ventana, de forma que se observen las variaciones de los elementos precedentes y subsecuentes. Las colocaciones se registran en un banco de colocaciones para tener acceso a los datos de los perfiles de colocación, de forma que el cambio semántico de una palabra se define con base en un cambio significativo entre su perfil de colocación registrado y su perfil común.

De esta forma, en combinación con los análisis diacrónicos de los corpus, se pueden encontrar variaciones de los perfiles de colocación de una unidad para poder determinar si el cambio que está ocurriendo a través del tiempo corresponde a un cambio semántico.

El tercer filtro es la identificación y extracción de nuevas palabras compuestas potenciales. Este filtro se puede considerar como un complemento del filtro anterior ya que también se basa en el uso de patrones de colocaciones para realizar el análisis, pero en este caso se analizan las combinaciones nuevas que tienen un nodo determinado apoyándose en el banco de patrones de colocaciones. Bajo este supuesto, las combinaciones podrían representar un nuevo sentido o una nueva palabra compuesta. Y finalmente el cuarto filtro consiste en la detección de nueva sinonimia.

En conclusión, la metodología se fundamenta en reglas lingüísticas, pero el sistema opera mediante reglas heurísticas. En combinación con el trabajo con corpus, este sistema permite reducir el trabajo manual de selección en gran medida, sin embargo, April aún requiere una edición manual posterior. Como se menciona en relación con la NS, April tiene dos problemas fundamentales: la definición superficial de novedad no distingue entre significado nuevo y referencia nueva, y tampoco puede identificar colocaciones que no se acercan al umbral de mínimo cuatro concordancias. Es decir, "no puede detectar las colocaciones raras y por tanto tampoco nuevos significados o referencias que aparecen raramente" (Renouf, 2010, p. 140).

A pesar de que el análisis de patrones de colocaciones da resultados alentadores, habría que considerar incluir medidas estadísticas más detalladas que permitan filtrar mejor los candidatos a neologismos. Sin embargo, también hay que considerar que April fue la primera herramienta especializada en la detección de NS.

2.2 Logoscope

Logoscope se desarrolló en la Universidad de Estrasburgo de 2012 a 2015. Es una herramienta informática que revisa páginas web de prensa cotidiana en francés (*Le monde*, *La Croix*, *L'Equipe*, *Dernières Nouvelles d'Alsace*, etc.) para extraer los artículos mediante RSS todos los días. De acuerdo con Falk et al. (2014b), esta aplicación tiene como finalidad adquirir neologismos de forma semiautomatizada para producir un repositorio dinámico que pueda ser usado por diferentes usuarios, y también detectar la primera aparición de cada neologismo.

Su metodología de trabajo se concentra en las condiciones textuales y discursivas de la innovación léxica, de forma que mediante el análisis de esta información se realice la

[4]En español: Mary tenía un cordero pequeño, su lana era blanca. En este caso el nodo sería *lamb* o *cordero* en español.

extracción de candidatos adecuados a neologismo (Gérard et al., 2014, p. 2627). Abordan la detección como un problema de clasificación basado en diferentes tipos de rasgos extraídos de los artículos que conforman el corpus de prensa. Estos rasgos sirven para analizar cuáles son más útiles para detectar candidatos a neologismos dentro de las diferentes temáticas del corpus.

Emplearon 2723 artículos como corpus inicial, este corpus fue segmentado y *tokenizado* con la herramienta TinyCC (Falk et al., 2014b). Para realizar el etiquetado evaluaron 7 etiquetadores o *taggers* en francés: LGtagger, SEM, LIA_tagg, Stanford Tagger, MElt, Talismane, y TreeTagger. Cada etiquetador fue puesto a prueba con el mismo corpus de referencia en formato XML para comparar la eficacia documentada contra eficacia real de cada etiquetador y así seleccionar el más adecuado para Logoscope, la lista de etiquetadores que fueron evaluados se puede ver en la tabla 2.1.

Etiquetador	Totales	Sust. (293)	Verb. (68)	Adj. (81)	Loc. (13)	Comp. (28)
LG tagger	73.30	82.08	72.06	43.04	66.67	0.00
LIA_tagg	72.17	79.93	66.18	51.90	66.67	0.00
MElt	83.26	92.83	67.65	64.56	75.00	91.67
SEM	67.42	81.36	50.00	36.71	58.33	62.50
Stanford	85.29	92.47	89.71	60.76	50.00	87.50
Talismane	81.45	97.85	54.41	48.10	66.67	79.17
TreeTagger	82.35	93.91	75.00	53.16	75.00	91.67
Mayoría	86.43					

Tabla 2.1 – Porcentaje de sustantivos (Nom.), verbos (Verb.), adjetivos (Adj.), locuciones (Loc.) y palabras compuestas (Comp.), etiquetadas correctamente (Falk et al., 2014c).

Se tomaron en cuenta las etiquetas y el entrenamiento previo, específicamente la segmentación de palabras. Como se observa en la tabla los etiquetadores que produjeron mejores resultados fueron Stanford Tagger (85.29 %), MElt (83.26 %) y TreeTagger (82.35 %), los demás etiquetadores tuvieron diversos problemas, tales como utilizar un juego de etiquetas demasiado complejas como LIA_tagger (Falk et al., 2014c). Tras este etiquetado, se detectaron 629 palabras desconocidas, de las cuales se seleccionaron manualmente 81 neologismos de forma.

Los rasgos explorados fueron: rasgos formales, morfoléxicos y temáticos. Estos últimos aportan información adicional no provista por los rasgos morfoléxicos y son indispensables para realizar el modelado de temas. Este acercamiento se fundamenta en la idea de que todo documento es una combinación de temas, donde un tema es la probabilidad de este mismo distribuida entre las palabras del corpus. Para poner a prueba estos supuestos primero se conformó un corpus de temas generales de periodismo empleando una compilación de diarios en línea y, basándose en los temas obtenidos, se estimó el contenido temático del corpus de prueba restringiendo la salida a 10 temas.

En el siguiente paso del procesamiento se usó una máquina de vectores de soporte con LibSVM[5] y Weka Toolkit[6] con la configuración predeterminada, y posteriormente se realizó una validación de precisión de 10 cruces, exhaustividad y *f-score*, para cada clase

[5] `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`
[6] `https://www.cs.waikato.ac.nz/ml/weka/`

por separado y para las dos clases tomadas en cuenta. Los neologismos son resultado directo del proceso de clasificación (ver figura 2.2) y las palabras plausibles son aquellas formas no conocidas que pueden ser palabras, pero no necesariamente neologismos. Como estos últimos elementos también son de interés, se clasificaron los resultados de verdaderos positivos usando la probabilidad de salida de LibSVM. Finalmente, se evaluó cuántos neologismos se detectaron correctamente (verdaderos positivos) entre los 81 neologismos formales que fueron seleccionados manualmente. Usando las técnicas de aprendizaje automático, las palabras desconocidas pueden ser presentadas de manera significativa para el neólogo. El resultado más relevante de la aplicación es el de la exhaustividad de la clase positiva, que refleja el número de neologismos reconocidos.

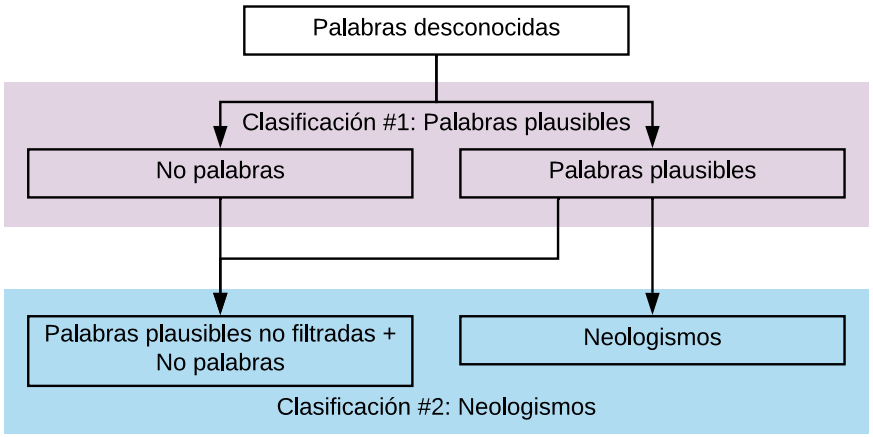


Figura 2.2 – Secuencia de clasificación (Falk et al., 2014b).

La combinación que da mejores resultados para la detección de neologismos es *forma+lex* para palabras posibles y *forma+lex+tema* para neologismos. El beneficio de usar modelos de temas es que da acceso al contenido semántico de palabras desconocidas, yendo más allá de las ventanas de coocurrencias. Con relación al tema, se observó que permite la detección de neologismos que carecen de propiedades predominantes. No obstante, los rasgos temáticos parecen favorecer palabras que no son plausibles considerando las reglas de formación de palabras tradicionales del francés.

Como se menciona en Gérard et al. (2014), la detección de NS es una de las posibles aplicaciones de Logoscope. La metodología de trabajo de esta aplicación aplica la teoría de *Topic Models* de Lau et al. (2013). Un *Topic Model* es un modelo probabilístico que permite determinar los sujetos o temas dentro de una colección de textos, en este caso, un corpus periodístico. Estos modelos son analizados como distribuciones de probabilidad entre el espacio de las palabras (Falk et al., 2014a). Las observaciones obtenidas de los análisis de temas realizados durante el diseño de Logoscope abrieron la posibilidad de implementar este método para detectar nuevos significados de palabras ya que el análisis de *Topic Models* realiza un modelado de contextos temáticos y analiza su evolución en el corpus.

Para poner a prueba esta hipótesis comprobaron si existe cambio semántico en la palabra *quenelle*[7] comparando dos subcorpus. El primer subcorpus (REF) registra el significado clásico de la palabra en el ámbito culinario —mencionado anteriormente— y el

[7] "Sorte de rouleau fait avec une farce pochée de poisson, de viande ou de volaille". En español: Tipo de rollo hecho con un relleno cocido de pescado, de carne o de ave de corral. Visto en: `http://www.laro`

segundo (NOUV) se compone por una combinación de 160 artículos de varios periódicos donde se espera observar el cambio semántico, en caso de existir. Para corroborar los significados de esta palabra se tomaron en cuenta las acepciones de *quenelle* que estaban registradas en diccionarios de referencia, los diccionarios Reverso, Larousse y TLFi registran el significado culinario y en el Wiktionary[8] y Wikipedia[9] se encuentra registrado un segundo significado de *gesto*. La hipótesis de trabajo es que el significado de *gesto* o *seña* no está registrado hasta 2005, por lo tanto se consideró que el corpus REF no contiene usos de *quenelle* con el significado de gesto, por el contrario el corpus NOUV podría contener el uso de ambos significados, el culinario y el de *gesto*.

Para realizar la clasificación de temas se usó el método *Hierarchichal Dirichlet Process* (HDP), mediante el cual se infirieron 6 temas diferentes T (ver tabla 2.2). Después de realizar un segundo análisis, la distribución de los temas resultó 56 % para T1 y 44 % para T2. Con estos resultados procedieron a confirmar si estos temas detectados correspondían al significado registrado de *quenelle* mediante los métodos de Lau et al. (2013) y Aletras et al. (2014).

T1	quenelle geste antisémite salut photo nazi bras humoriste français effectuer...
T1	quenelle jean marier marque fion fond glisser petite sionisme déposer...
T3	quenelle carte menu pat brochet cuisine rue compter veau dessert...
T4	quenelle spectacle public geste jeune interdiction antisémite humoriste monde ordre...
T5	quenelle brochet pain épices grand sauce chair cuisine écrevisse beurre...
T6	vide quenelle couverts charges cru debout derrière docteur effet émission...

Tabla 2.2 – Lista de las 6 temáticas detectadas con HDP (Falk et al., 2014a).

El primero fue *correspondencia de temas* $\leftrightarrow$ *uso en el corpus*. Este método presenta el número de párrafos en los cuáles un tema determinado es predominante, es decir, muestra cómo se distribuye cada sentido en cada corpus en este caso los T1, T2 y T4 que hacen referencia al sentido de *gesto* se registraron de forma predominante en el corpus NOUV. Posteriormente analizaron el método de *correspondencia de sentidos* $\leftrightarrow$ *tema* que emplea la ecuación 2.1 para calcular la similitud con la información registrada en los diccionarios de referencia. Esta función calcula la similitud de un significado del diccionario s_i y de un tema t_j teniendo en cuenta que S y T son las distribuciones multinomiales de las palabras de cada significado s_i y tema t_j, respectivamente, $M = \frac{1}{2}(S + T)$ y $JS(X||Y)$ y $KL(X||Y)$ son las divergencias de las distribuciones X e Y, respectivamente.

$$sim(s_i, t_j) = 1 - JS(S||T) = 1 - \frac{1}{2}KL(S||M) - \frac{1}{2}KL(T||M) \tag{2.1}$$

usse.fr/dictionnaires/francais/quenelle/65629

[8]"(Néologisme) (France) Sorte de bras d'honneur consistant à placer sa main ouverte sur le haut du bras opposé et tendu vers le bas". En español: Neologismo, Francia, Tipo de seña que consiste en colocar la mano abierta sobre el alto del brazo opuesto y extendido hacia abajo. Visto en: https://fr.wiktionary.org/wiki/quenelle

[9]"[...]consiste à tendre un bras vers le bas tout en posant la main de l'autre bras sur l'épaule". En español: Consiste en extender un brazo hacia abajo y colocar la mano del otro brazo sobre el hombro. Visto en: https://fr.wikipedia.org/wiki/Dieudonne

La ecuación 2.2 corresponde al segundo análisis utilizado, *significado predominante*, que calcula cuál de los significados detectados tiene mayor presencia en la totalidad del corpus. Tomando como base los resultados de las similitudes antes calculadas T es el número de temas totales y $f(t_j)$ es el número de párrafos que contienen el uso del tema predominante t_j:

$$\text{prevalence}(s_i) = \sum_{j=i}^{T} \left(\text{sim}(s_i, t_j) \times \frac{f(t_j)}{\sum_{k=1}^{T} f(t_k)} \right) \tag{2.2}$$

Finalmente para obtener el índice de novedad $\text{Nouv}(t_j)$ para cada tema t_j se emplea la ecuación 2.3, donde $p\text{NOUV}(t_i)$ y $p\text{REF}(t_i)$ son las probabilidades de los temas t_i dentro de los corpus NOUV y REF, respectivamente. Siguiendo este proceso, un resultado elevado indicaría una frecuencia elevada de un tema en el corpus NOUV y simultáneamente baja frecuencia en el corpus REF. En este caso los temas T1, T2 y T4 resultaron predominantes y se confirmaron como nuevos significados de *quenelle* correspondientes a *gesto*.

$$\text{Nouv}(t_i) = \frac{p\text{NOUV}(t_i) - p\text{REF}(t_i)}{p\text{REF}(t_i)} \tag{2.3}$$

El rol de las diversas variables[10] y de las herramientas en las que se apoya Logoscope es fundamental para llevar a cabo los análisis de neologicidad y determinar la pertenencia de un elemento a una temática determinada. En particular es interesante el tratamiento de la variable temática (ver figura 2.3), ya que no es considerada como una variable textual que no siempre es documentada. Esta variable es analizada desde dos puntos de vista, el léxico y el contextual. El primero denota el dominio al cual pertenece un neologismo, y el segundo permite (de la misma forma que en plataformas informáticas como Wortwarte) mostrar los neologismos pertenecientes a una temática en concreto.

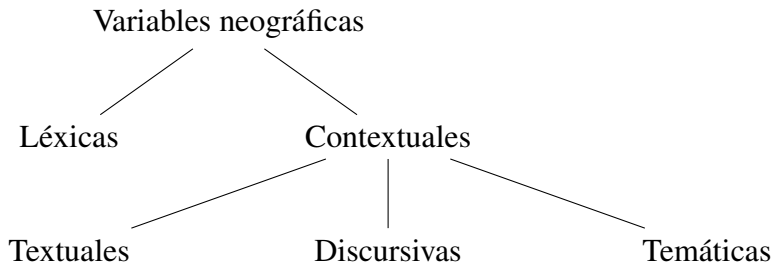


Figura 2.3 – Arquitectura de las variables neográficas (Gérard et al., 2014, p. 2629).

La metodología de Logoscope se fundamenta en los siguientes principios:

- Análisis ya existentes en materia de detección y extracción de neología.
- Tratamiento automático de la lengua.

[10]La variable *temática* no se incluye en el diagrama original, pero se desarrolla como variable contextual posteriormente.

- Segmentación temática.

- Uso de máquinas de vectores de soporte (SVM) para la detección de temas y clasificación automática de documentos.

El uso de métodos de aprendizaje automático supervisado es un elemento en común entre DENISE y el propuesto por Gérard et al. (2014). Sin embargo, el tipo de métodos empleados por Logoscope es diferente ya que Logoscope emplea métodos de clasificación, mientras que DENISE utilizará una combinación de métodos de aprendizaje automático y representaciones vectoriales de palabras (ver sección 1.5).

Para detectar los candidatos a neologismos Logoscope emplea un clasificador automático basado en ejemplos previamente etiquetados. Con la información etiquetada se determinaron tres tipos de rasgos: rasgos relativos a la forma de la palabra, rasgos morfoléxicos y rasgos relativos al contexto temático textual. Este último rasgo es el que permite determinar a qué temática pertenece un candidato a neologismo.

El tratamiento del tema es el aspecto fundamental en la metodología de Logoscope ya que la hipótesis que se plantea para detectar NS es que un candidato a NS se encuentra en una temática distinta a las que se encontraría con regularidad. Como se menciona en el ejemplo de *quenelle*, que aparecía con regularidad en el ámbito culinario como un platillo tradicional francés y se encontró en el ámbito de la política como una seña o gesto que se hace con la mano.

Esta clasificación se hizo con la ayuda de Mallet[11] y ha servido para inferir un *arreglo* o vector que contiene las 10 temáticas mostradas en la figura 2.4. Para agrupar las palabras relacionadas con cada temática emplearon Termite[12] (Gérard et al., 2014, pp. 2636-2638). Con este programa se llevó a cabo la clasificación automática de 629 palabras no conocidas usando un método de clasificación supervisada, en este caso, una SVM. Un primer análisis manual determinó que 81 candidatos podrían considerarse neologismos verdaderos. Los mejores resultados se obtuvieron con el análisis del tema tal como se muestra en la tabla 2.3.

Clase	Precisión	Exhaustividad	f-Score	Neo. Verdaderos
Positivo	0.129	0.889	0.225	
Positivo y Negativo	0.844	0.295	0.338	72

Tabla 2.3 – Resultado de los experimentos de clasificación de tema (Gérard et al., 2014, p. 2640).

Esta metodología parece prometedora, sin embargo, se menciona en Falk et al. (2014a) que aún resta validar los resultados para los otros significados documentados y encontrar las temáticas a las que pertenecen. También cabe mencionar que otro de los problemas que Logoscope presenta es: "L'un des points problématiques de la méthode réside dans l'alignement des topics avec les définitions trouvées dans dictionnaires: les répertoires de sens et la granularité son fortement dépendants du dictionnaire utilisé" (p. 7).

[11] `http://mallet.cs.umass.edu`
[12] `http://vis.stanford.edu/papers/termite`

Mientras que el uso de fuentes lexicográficas desde un enfoque manual es un paso lógico para la detección de neologismos, desde una perspectiva computacional la composición de las definiciones del diccionario es una limitante. Tal como los autores mencionan, la granularidad del significado dependen en gran medida del diccionario que está siendo empleado. Como ejemplos de esta limitante podemos mencionar las siguientes particularidades: definiciones tautológicas, definiciones sinonínimicas, referencias a otras entradas o definiciones, etc.

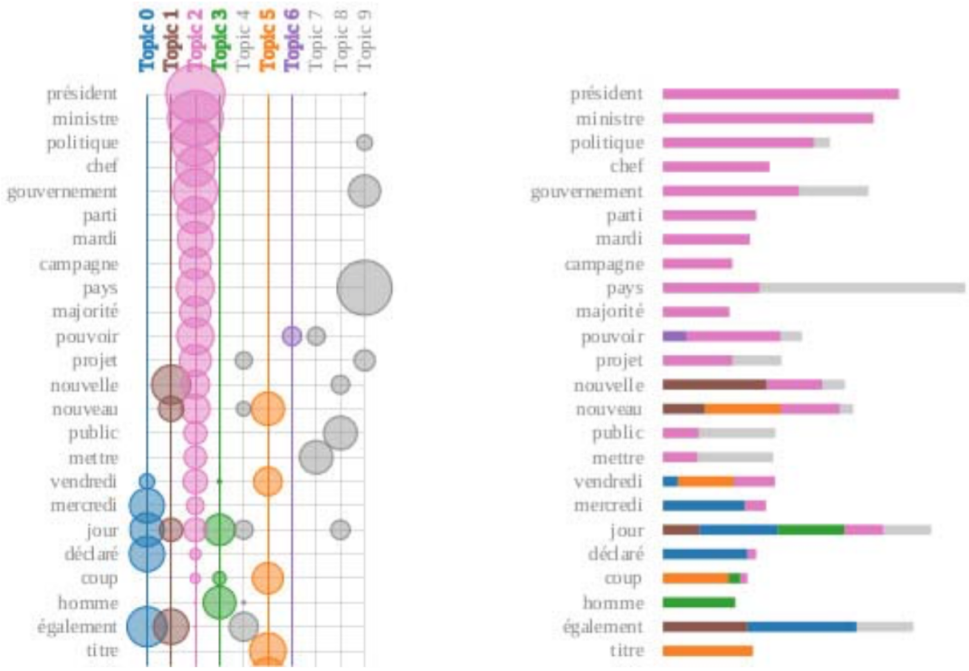


Figura 2.4 – Temáticas obtenidas con Termite (Gérard et al., 2014, p. 2638).

Las diferencias entre Logoscope y DENISE radican en la aplicación de distintos métodos de aprendizaje automático, así como la finalidad de su implementación. Logoscope usa, principalmente, una SVM para realizar la clasificación de los candidatos a neologismo al compararlos con un tema predefinido (obtenido de una clasificación previa) y para realizar la extracción de las palabras claves pertenecientes a estos temas. La finalidad de Logoscope es agilizar el proceso de clasificación y extracción para que el neógrafo realice su tarea de forma más efectiva, con la posibilidad de detectar NS.

Por otra parte DENISE se centrará en la detección y extracción de NS y no de otros tipos de neologismos. También tratará la detección de NS como un problema de clasificación, pero empleando la desambiguación de temas entre las concordancias que serán analizadas y los campos semánticos de los posibles candidatos que fueron extraídos de dicha concordancia. Bajo esta hipótesis de trabajo, existe un modelo con temáticas previamente modeladas y un modelo de lengua que servirá como un repertorio de referencia de la lengua general.

2.3 Otras aproximaciones

Algunos trabajos como los de Nazar et al. (2007), Nazar (2010, 2011, 2013) y Janssen (2005a,b, 2009, 2012) presentan diferentes estrategias metodológicas para la extracción de terminología y neología. Nazar aborda la aplicación de estas metodologías para NS, pero aún no se ha desarrollado una aplicación destinada a esta tarea específica. La primera propuesta metodológica de Nazar (2010) surge en respuesta a las limitaciones que presenta el criterio lexicográfico cuando se trata de detectar NS.

2.3.1 Acercamientos metodológicos de Rogelio Nazar

Nazar destaca dos problemas fundamentales: los NS aparecen con otro significado en el corpus de exclusión y el vaciado manual sigue siendo indispensable para determinar con certidumbre la existencia de un NS. Para dar solución a estos problemas el autor propuso un estudio de las concurrencias, entendiendo concurrencia como la aparición de dos unidades a una distancia flexible y sin importar el orden de aparición dentro de una ventana de contexto de *n* palabras. El comportamiento de esta unidad debería variar con el tiempo, es decir, cuando una palabra *X* con un significado conocido se asocia con concurrentes diferentes a los de concurrentes del significado conocido, este nuevo grupo de concurrencias podría indicar la existencia de una nueva *X'* y por ende un probable NS. Por lo tanto, la hipótesis sería que en el caso de los NS el cambio de perfil de concurrencia de una unidad denota un nuevo significado.

Los experimentos se llevaron a cabo con un corpus formado por los archivos del periódico *El País* desde 1976 hasta 2007, un periodo de 31 años. Para la extracción automática de neologismos se propuso comparar mediante la distancia euclidiana (ver ecuación 2.4) entre dos puntos $P_1 = (X_1, X_2, \ldots, X_n)$ y $P_2 = (Y_1, Y_2, \ldots, Y_n) \in \mathbb{R}^n$ y la curva proyectada por un neologismo ideal representada por la función 2.5 donde $f(x)$ es igual al periodo de tiempo que abarca el corpus. En este plano el eje *x* representa los años y el eje *y* frecuencia relativa de la unidad. La distancia euclidiana por lo tanto determinará si existe similitud entre el comportamiento del candidato a neologismo y la curva del neologismo ideal, un comportamiento similar indicaría un candidato válido a neologismo.

$$d_E(P_1, P_2) = \sqrt{\sum (X_i - Y_i)^2} \tag{2.4}$$

$$f(x) = x^{10} \tag{2.5}$$

Según Nazar (2010), los rasgos que permiten la distinción del significado de un NS se encuentra en el contexto. Así que se realizó un análisis de histogramas, donde el candidato a neologismo ocupa la posición cero, también llamada nodo, en el eje horizontal y en el eje vertical la frecuencia de cada palabra asociada a este nodo. Debido a este comportamiento se aplicó un algoritmo de *clustering* (Nazar et al., 2007) para identificar los cambios de perfil en el contexto. Este *clustering* se realiza después de eliminar las palabras con mayor frecuencia, y por lo tanto menos informativas, y las agrupa por similitud morfológica para tener una sola forma y así aumentar la frecuencia relativa de estas palabras.

Cada nodo es un apuntador a las concurrencias de la unidad en cuestión. Con la ecuación 2.6 se ponderan los arcos que existen entre nodos, en este caso *x* es el candidato e *i* y *j*, los nodos que ocurren con el candidato, mientras que N representa el número total de contextos donde aparece x. El resultado que se obtiene (ver tabla 2.4) es una agrupación de los documentos donde ocurren contextos similares. En este caso se analizaron dos agrupamientos: *empeñar* y *escotes*.

$$R_{i,j}(x) = \log\left(\frac{F_{ij}(x)}{N}\right) \tag{2.6}$$

En el ejemplo se observan dos agrupamientos para *palabra de honor*, el sentido ya conocido bajo el nodo que tiene mayor cantidad de arcos *empeñar* y un grupo nuevo bajo el nodo *escote*. Esta metodología agrupa los términos que tienen mayor cantidad de arcos y ordena los documentos de forma cronológica para observar la evolución del uso de *palabra de honor* detectando así un contexto con nuevas concurrencias y, por lo tanto, un contexto diferente al ya conocido.

Agrupamiento 1: empeñar		Agrupamiento 2: escotes	
Términos	Contextos	Términos	Contextos
1) ap	1) 1979420.txt	1) copresidente	1) 19981019.txt
2) astarloa	2) 19810501.txt	2) cubren	2) 20020203.txt
3) barrionuevo	3) 19850524.txt	3) drapeados	3) 20020930.txt
4) confederal	4) 19850913.txt	4) escotes	4) 20070118.txt
5) consentido	5) 19889331.txt	5) gucci	5) 20071201.txt
6) credulidad	6) 19970520.txt	6) marrón	
7) cuan	7) 19970814.txt	7) modista	
8) empeñar	8) 19980908.txt	8) ojito	
9) esclarece	9) 20041119.txt	9) organza	
10) escudero		10) swarovski	
11) fusté		11) tonos	
12) herrero			
13) incité			
14) inocencia			
15) proclamar			
16) quebrantamiento			
17) reiterado			
18) tejero			

Tabla 2.4 – Agrupamientos de concurrencias de *palabra de honor* (Nazar, 2010).

El sustento teórico de esta metodología basada en análisis de concurrencias es analizar la NS como un fenómeno diacrónico de cambio semántico y, por tanto, como una rama particular de la lingüística diacrónica donde existe ya una tradición y una clasificación de tipos de cambio de significado. En Nazar (2011) se presentan antecedentes metodológicos de los análisis de grafos que servirán de sustento para la metodología propuesta anteriormente en Nazar (2010).

En Nazar y Cabré (2012) los autores expanden las ideas anteriores, mientras que el artículo trata de una propuesta de un sistema de extracción de terminología y no de neología, los métodos de aprendizaje estadístico asistido propuestos son retomados en Nazar (2013)

como parte del procesamiento de los datos. El algoritmo se entrena con textos terminológicos obtenidos de un corpus LSP (ejemplos positivos de términos) y con contextos de lengua general (ejemplos negativos de términos) para desarrollar un modelo estadístico que represente los rasgos principales de ambas muestras.

Después de realizar el etiquetado POS de los elementos del corpus LSP, el algoritmo está listo para recibir una entrada de datos y generar una lista de candidatos a término. Con dicha lista de candidatos y sus frecuencias relativas en cada uno de los corpus, se calcula la frecuencia de los atributos de cada nivel i por cada candidato a término $T(c)$. Tal como se muestra en la ecuación 2.7, $f_o(C_i)$ representa la frecuencia relativa observada de un atributo i en el corpus de entrenamiento, mientras que $f_e(c_i)$ es la frecuencia relativa de dicho elemento en el corpus de referencia y $f_a(c)$ es la frecuencia real de c dentro del texto analizado. Por lo tanto, una unidad es considerada candidato a término cuando la frecuencia relativa de dicha unidad es mayor en el corpus LSP, en comparación con su frecuencia relativa en el corpus de lengua general.

$$T(c) = \left(\prod_{i=1}^{|c|} \frac{f_o(c_i)}{f_e(c_i) + 1} \right) f_a(c) \tag{2.7}$$

Los resultados que se obtuvieron fueron de 85 % de precisión con las primeras 200 palabras en comparación con los otros algoritmos puestos a prueba, listas de bigramas y chi-cuadrada. Si bien este artículo no trata estrictamente sobre neología, es una clara muestra de los buenos resultados obtenidos de la combinación de técnicas de aprendizaje asistido y estadística. Las estrategias propuestas por Nazar para detectar NS han sido documentadas, pero aún no se ha desarrollado un programa que realice esta tarea. Actualmente su enfoque es primordialmente WSI y grafos, pero es claro que la combinación de métodos, seleccionando los más eficientes en cada paso de la metodología, proporciona los mejores resultados.

En Nazar (2011) se expande el método cuantitativo analizado en Nazar (2010), la detección de NS mediante análisis de concurrencias. El autor presenta de forma más detallada el sustento teórico de su proyecto con resultados preliminares empleando un corpus de trabajo diferente, el corpus *Google Books Ngrams*[13]. Comienza proponiendo que las estrategias de detección de NS no pueden ser las mismas que las estrategias para detectar neología formal. Sin embargo, afirma que toda estrategia de detección de neologismos es en esencia una forma de clasificación modificable al encontrarse frente a nueva información. Esto deja abierto la posible combinación de estrategias para obtener mejores resultados. Esta propuesta menciona cuatro estrategias metodológicas en materia de detección de NS:

- Empíricas (Gross, 1994; Hanks, 2004; Mejri, 2006, 2010).
- Basadas en coocurrencias (Collier, 1998; Sagi et al., 2009; Cook y Stevenson, 2010; Renouf, 2010).
- Basadas en *clustering* (Holz y Teresniak, 2010; Boussidan y Ploux, 2011; Rohrdantz et al., 2011).

[13] `https://books.google.com/ngrams`

- Basadas en análisis de bigramas (Cook y Hirst, 2011; Gulordava y Baroni, 2011; Renau y Nazar, 2011).

Estos trabajos son antecedentes metodológicos que han servido al autor para diseñar una estrategia basada en la implementación de *clustering* en combinación con grafos de concurrencia que servirán para realizar la inducción de significado (WSI), esto es agrupar contextos de aparición de una determinada unidad léxica de acuerdo con criterios de similitud en el vocabulario de los contextos, además de analizar las relaciones sintagmáticas entre palabras mediante las frecuencias de concordancia (Nazar, 2011, p. 17-19). Esta estrategia será el foco principal de este estudio y el punto de partida para un análisis más complejo en Nazar (2013).

Para realizar el análisis, Nazar, parte de la siguiente pregunta: "¿Es posible detectar el cambio semántico a través del cambio de combinatorias de las unidades analizadas?". Entendiendo como combinatoria al grupo de palabras que suelen aparecer junto con la unidad analizada (o candidato a NS), es decir, una combinatoria distinta a las ya registradas indicaría un cambio de sentido. El periodo total que abarca el corpus es de 1975 a 2008, y fue dividido en dos subperiodos, el primero de 1975 a 1995 y el segundo de 1996 a 2008. Estos dos periodos sirvieron para realizar la extracción de los n-gramas en los cuales aparece el candidato, así como sus frecuencias.

La metodología que se siguió consistió en estudiar las palabras que ocurren en conjunto con cada verbo de una lista de 13 verbos del dominio de la informática. Para ello, se asignó un índice de neologicidad a cada verbo, calculado mediante la fórmula 2.8, donde $j_{i,1}$ sería la frecuencia de las colocaciones de i del verbo j durante el primer periodo de tiempo analizado (1975-1985) que fueron separados del periodo de un segundo periodo de tiempo (1986-1995) para visualizar resultados.

$$\mathrm{Neo}(j) = \sum_{i=1}^{|j|} \frac{j_{i,3}}{j_{i,1} + j_{i,2} + 1} \tag{2.8}$$

Para realizar la evaluación de esta metodología se mezclaron los 13 verbos con otros verbos seleccionados de forma aleatoria para obtener un total de 250 combinaciones verbos para analizar (Nazar, 2011, p. 19-21). En la tabla 2.5 se muestran las primeras 25 posiciones donde 8 de los 13 verbos neológicos analizados aparecen en las primeras 20 posiciones.

Los resultados obtenidos son relativamente buenos, ya que en las primeras 50 posiciones aparecen 11 de los 13 verbos, representando 86.60 % de exhaustividad y 22 % de precisión. Para validar la significancia de estos resultados se usó la prueba de chi-cuadrado, ver ecuación 2.9). Dada una población, donde se analiza un carácter X con $(x_1, x_2, \ldots, x_n)$ modalidades excluyentes, denotando por O_i la frecuencia observada de x_i y por E_i la frecuencia esperada o teórica de x_i, se define

$$X^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{2.9}$$

Tras su aplicación se obtuvo un resultado de $\alpha = 34{,}308$, con cuatro grados de libertad que equivale a un valor de *p* inferior a 0.001, es decir, una muy baja probabilidad de que estos resultados hayan ocurrido de forma aleatoria (Nazar, 2011, p. 23).

Rango	Verbo	Valor	Lista Inicial
1	activar	7394.91	
2	actualizar	3251.33	✗
3	arrastrar	1914.5	✗
4	abreviar	1689	
5	existir	1650.25	
6	visitar	1582.5	✗
7	añadir	1373.25	
8	copiar	1348.83	✗
9	quitar	1345.08	
10	mantener	1097.92	
11	construir	1042.5	
12	combinar	853.5	
13	afiliar	831	
14	amoldar	672	
15	ejecutar	611.33	✗
16	pegar	597.75	✗
17	detener	581.25	
18	vincular	401.5	✗
19	almacenar	372.5	✗
20	revelar	355.25	
21	descargar	315	✗
22	propiciar	307.75	
23	cifrar	269.5	
24	dialogar	199.67	
25	aclarar	181	

Tabla 2.5 – Fragmento del análisis de las combinaciones de verbos neológicos (Nazar, 2013, p. 22).

En conclusión, este estudio presenta un ejercicio más detallado que su antecesor. No obstante, el autor propone a futuro afinar el análisis implementando lematización, etiquetado morfosintáctico, *chunking* para la detección de unidades sintagmáticas y análisis de dependencias para estudiar la relación entre predicados (Nazar, 2011, p. 27), ya que las características del corpus empleado en este experimento no permitieron realizar ninguno de estos procesos.

Como continuación de estas propuestas, Nazar (2013) se centra en resolver un problema que tanto lingüistas como especialistas de otras áreas del conocimiento (filosofía, filología, etc.) han enfrentado durante años: poder determinar cuándo un texto hace referencia a otras entidades que comparten el mismo nombre, mediante un análisis de frecuencias de las palabras que son encontradas en el contexto de aparición de la palabra objetivo.

Para este fin primero realiza una diferenciación conceptual. Define *Word Sense Disambiguation* (WSD) como la operación mediante la cual un autómata asigna un sentido determinado a una palabra ambigua en contexto de un inventario de sentidos disponible en el sistema. Y lo contrasta a *Word Sense Induction or Discrimination* (WSI), que es la operación de encontrar esos sentidos de una muestra de contextos de ocurrencias de una palabra dada (Nazar, 2013, p. 8).

El tema central de este enfoque principal son los algoritmos de *clustering* basados en gráficos de ocurrencias que pueden ser útiles para resolver estos tipos de ambigüedad semántica. El modelo propuesto por el autor produce para cada *input* o entrada, una re-

presentación de las relaciones sintagmáticas de dicha palabra con las frecuencia de las palabras que aparecen en su contexto. Cada palabra polisémica tiene su propio grupo de *palabras amigas* de acuerdo con el principio de Firthian (Firth, 1957). Estas *palabras amigas* son la base que sirve para determinar a cuál de las acepciones del término en cuestión corresponde un contexto de aparición de dicho término.

Para Nazar (2013) la diferencia más significativa entre los enfoques vectoriales y los gráficos de coocurrencias, es que estos últimos son más prometedores para el WSI y el WSD, ya que el método propuesto es más simple conceptualmente y también a nivel computacional, y además descarta las fuentes de conocimiento externas, tanto lingüísticas como ontológicas. La simplicidad conceptual no es muy diferente a los acercamientos de Molina et al. (2010) y Gérard et al. (2014): un término cuyo contexto de aparición sea semántica o temáticamente diferente a uno de sus contextos definitorio ya conocidos da indicio de un término que tiene un contenido semántico nuevo.

La metodología emplea los siguientes cálculos. Primero usa el coeficiente de Dice entre dos muestras I y J (ver ecuación 2.10) para realizar una *pseudo-lematización* basándose en la similitud ortográfica de las palabras o en la similitud de bigramas.

$$\mathrm{Dice}_{(I,J)} = \frac{|I \cap J|}{|I| + |J|} \tag{2.10}$$

El siguiente paso es mantener las unidades que aporten más información y descartar las unidades que aportan menos para reducir la cadena de caracteres, esto se logra empleando el punto de información mutua (ecuación 2.11[14]) entre dos muestras I y J. Este proceso cuantifica la diferencia entre la probabilidad de su coincidencia dada su distribución conjunta $P(X, Y)$ y sus distribuciones individuales $P(X)$ y $P(Y)$, suponiendo independencia matemática.

$$I(x, y) = \log_2 \frac{P(x, y)}{P(x)P(y)} \tag{2.11}$$

Por último, las siguientes dos fórmulas, 2.12 y 2.13, realizan el proceso de desambiguación. La primera fórmula sirve para realizar un filtrado adicional y generar los grafos que, mediante un proceso de *clustering*, agruparan cada conjunto en función de su similitud, donde A_i y A_j son dos acepciones distintas de la acepción de una palabra, N representa el número total de acepciones de esa palabra.

$$R(A_i, A_j) = \frac{(A_i, A_j)}{N} \tag{2.12}$$

La segunda ecuación usa como entrada las palabras que componen los agrupamientos I y J para crear un nuevo agrupamiento con cada nueva asociación de palabras (i, j). Finalmente, con estas nuevas asociaciones de significados y agrupamientos, se crean nuevos agrupamientos más grandes. Por lo tanto, si una palabra tiene dos acepciones diferentes, dicha palabra debería generar dos agrupamientos distintos, cada uno con las palabras re-

[14]Nazar (2013) emplea la notación $\log 2$ en su obra, mientras que Church y Hanks (1990), la fuente original citada, emplea la notación $\log_2$, por lo tanto hemos optado por mostrar notación propuesta por la fuente original.

lacionadas correspondientes a cada acepción.

$$\text{overlap}_{(i,j)} = \frac{|I \cap J|}{\text{mín}(|I|, |J|)} \tag{2.13}$$

Los resultados son alentadores; de acuerdo con el autor, la evidencia sugiere que los patrones de coocurrencia son una propiedad del lenguaje en general y que una de las aplicaciones a futuro podría ser la detección de NS, tratándolos como un caso especial de polisemia. Este acercamiento comparte algunos puntos metodológicos, sin embargo, el proceso de inducción es diferente al propuesto en este proyecto de tesis, por lo tanto habría que realizar una comparación de resultados entre las dos metodologías.

2.3.2 Acercamientos metodológicos de Maarten Janssen

Maarten Janssen desarrolló NeoTag[15], un sistema que tiene como función principal la detección semiautomática de neologismos sintácticos, a diferencia de NeoTrack (sistema actualmente descontinuado) el cual estaba enfocado principalmente en la detección de neología formal mediante reglas de exclusión (Janssen, 2005a). No obstante, NeoTag también puede ser usado para detectar neología formal y, como función secundaria, puede emplearse como un etiquetador convencional que presenta ventajas sobre otros etiquetadores que funcionan fuera del contexto de la neología sintáctica (Janssen, 2012).

La metodología de detección de neologismos empleada por NeoTag consiste en comparar el porcentaje de certidumbre de cada etiqueta, es decir, aquellos índices más altos son elementos ya conocidos y por lo tanto no neológicos. Por otro lado, las *Deviantly Tagged Words* (DTW) o palabras con etiquetado ambiguo, son palabras tentativamente pertenecientes a otra categoría y, por lo tanto, candidatos a neologismos sintácticos. Entre esta categoría de DTW pueden existir discrepancias dado que dentro de este grupo podría contener palabras que no sean neologismos sintácticos o palabras que no sean DTW, es decir, que no ajusten a ninguna de estas dos categorías.

Para resolver estas discrepancias se comparan los candidatos obtenidos con una base de datos léxica OSLIN (Janssen, 2005b), filtrando así los resultados, confiando en que esta cuenta con una gran cantidad de información léxica. Finalmente, después de seguir estos procesos, se obtienen los candidatos finales con 97 % de precisión de clasificación antes de pasar por el filtro final y 70 % después de descartar los pares ambivalentes.

NeoTag es una herramienta que cuenta con varias funciones, puede ser utilizada como etiquetador, lematizador, detector de neología formal y detector de neología sintáctica. Esta herramienta considera la detección de NS como un subproducto de la detección de neologismos sintácticos (Janssen, 2012, p. 2124) tras un proceso de desambiguación por reglas estadísticas. El acceso libre permitía que cualquier persona interesada en realizar una investigación relacionada con este tipo de neología pueda contar con una herramienta que ha sido desarrollada y puesta a prueba bajo lineamientos actuales apegados a una metodología estricta.

[15] `http://marke.upf.edu/neotag/`, no disponible desde Abril de 2018.

2.4 Resumen de metodologías

La tabla 2.6 muestra el resumen de las estrategias empleadas por los cuatro enfoques que mostramos en los apartados anteriores. Pueden observarse puntos en común, como el tipo de interacción que tienen con el usuario y el uso de métodos estadísticos. Cabe mencionar que estos métodos han evolucionado con el paso del tiempo, ya que contar con un poder de procesamiento mayor permite llevar a cabo análisis más complejos.

	April	**Logoscope**	Nazar	Janssen
Lengua	Inglés	Francés	Español	Multilingüe
Tipo	En línea	En línea	Escritorio	En línea
Interacción	Semiautomática	Semiautomática	Semiautomática	Semiautomática
		Metodología		
Reglas estadísticas	✗	✗	✗	✗
Clustering			✗	✗
SVM		✗		
WSD		✗		
WSI		✗	✗	
Patrones	✗			
Índice de Dice			✗	
Topic Models		✗		

Tabla 2.6 – Comparación de estrategias usadas por April, Logoscope, Nazar (2010) y Janssen (2005a,b, 2012) para la detección de NS.

Capítulo 3

MARCO TEÓRICO

El siguiente apartado contiene la descripción de las bases teóricas en las que se enmarca el presente trabajo de tesis, siendo el eje teórico principal el concepto de neologismo semántico (NS) y los secundarios los conceptos de unidad terminológica o término y neología especializada o neonimia. Partimos de una cronología que presenta las principales definiciones del concepto de NS a lo largo de los últimos treinta años. Estas obras contextualizan al lector y muestran la ruta que ha servido como referencia y guía para el diseño de tipologías y propuestas teóricas contemporáneas sobre neología.

A partir de esta recapitulación procedemos a comparar dos tipologías contemporáneas de neologismos, así como la definición de nuestro objeto de estudio dentro de cada una de estas posturas. Esta comparación nos permitirá, por una parte, obtener una visión general del objeto de estudio en un contexto actual y, por otra parte, justificar por qué nos enmarcamos dentro de la propuesta teórica de Cabré (2009, 2011b), las obras que se derivan de esta visión y la metodología de trabajo del Observatori de Neologia de la Universitat Pompeu Fabra.

Desde esta perspectiva teórica–aplicada procedemos a presentar la segunda noción clave de esta tesis: la neología terminológica. Este tipo de neología es clave para este proyecto, puesto que uno de los supuestos metodológicos consiste en analizar la NS que surge desde un campo especializado hacia la lengua general. Por lo tanto, dentro de la NS, nos interesa delimitar estas unidades a aquellas que pueden considerarse neonímicas.

El tercer elemento teórico de esta tesis es el aspecto terminológico y, en concreto, la delimitación de unidades terminológicas. Estas unidades tienen características específicas que pueden ser delimitadas para llevar a cabo la extracción automática. Una de ellas, clave para nuestro trabajo, es el tema: identificar la temática de origen de un término nos puede ayudar a determinar si ha ocurrido un cambio de significado, ya que la nueva temática de esta unidad extiende un repertorio conceptual distinto al repertorio conceptual del significado preexistente en la lengua general.

3.1 Breve cronología sobre el concepto de neologismo semántico

A principios de los años setenta, Corbeil (1971) realizó un análisis de palabras nuevas —llevado a cabo durante un curso escolar— cuya finalidad fue determinar los principales

problemas que presentaba la neología. Como delimitación de las fuentes de origen de los neologismos, el autor propone la siguiente clasificación que incluye los NS bajo la categoría de extensión de significado (en francés, *extension de sens*):

- Composición.
  - Por derivación.
    - Derivación por prefijación.
    - Derivación por sufijación.
  - Por yuxtaposición.
  - A partir de raíces grecolatinas.
  - A partir de siglas.
- Préstamo.
  - Préstamo interno.
  - Préstamo externo.
- Creación *ex nihilo*.
- **Extensión de significado**.
- Cambio de categoría gramatical.

Corbeil (1971, p. 135) no aporta una definición formal de NS, pero sí ejemplifica este fenómeno comparando el significado registrado en el diccionario de la palabra *copieur* (relativo a una persona) con su significado actual (referido a una fotocopiadora).

Durante ese mismo año, en un estudio realizado por Guilbert (1971) sobre la neología científico–técnica, se explica que el lenguaje terminológico favorece la creación léxica en una lengua gracias, por un lado, a que científicos, técnicos e investigadores tienen la necesidad de crear nuevas palabras que den cuenta de la realidad de un campo determinado de la ciencia y, gracias, por otro lado, a la internacionalización de las ciencias. La siguiente tipología de neologismos muestra, según Guilbert (1971, p. 45), las categorías de neologismos que concentran la creatividad léxica en el dominio especializado:

- Neología por creación de bases inéditas.
- Neología por préstamo del modelo grecolatino.
- Neología por préstamo de términos extranjeros.
- **Neología semántica**.
- Neología sintagmática.
- Neología derivacional.

En lo que concierne a los NS, Guilbert (1971) explica que este fenómeno ocurre en dos direcciones: "par la spécialisation dans les vocabulaires particuliers de termes de la langue commune et par la migration de termes techniques et scientifiques dans le vocabulaire général" (Guilbert, 1971, p. 49). Por lo tanto, en este proceso bidireccional se consideran NS tanto las palabras que adquieren un significado especializado como los términos que se introducen en la lengua general; no obstante, solamente en el primer caso se puede hablar de *neología técnica*. Asimismo, existen dos procesos principales más que permiten la creación de NS:

- Emplear los términos preexistentes que, por su contenido semántico, cuentan con un carácter *indeterminado* y que requieren ser especificados dentro de la actividad particular donde se introducen, para así extender sus rasgos semánticos.
- Usar préstamos de un campo especializado e introducirlos dentro de un campo distinto. Los dos casos principales corresponderían al uso de procesos, métodos y técnicas preexistentes que son aplicadas en un campo distinto al usual, y procedimientos y técnicas nuevas que hacen uso de conceptos preexistentes como modelo de innovación léxica.

Dos años más tarde, Guilbert (1973) establece cinco postulados, creados a partir de la observación del funcionamiento de la lengua general, para determinar los fenómenos que permiten cada tipo de neología. Estos postulados son los siguientes:

- Una lengua funciona según su propio código en virtud del cual son producto los actos de discurso y de formación léxica.
- Un neologismo es un signo lingüístico formado por un significante y un significado.
- Un neologismo es una unidad de significación mínima, que es resultado de la combinación de elementos simples existentes dentro de la lengua.
- La creación de un neologismo no puede ocurrir sin estar asociada al discurso de los individuos/creadores que forman parte de una comunidad.
- Los neologismos presentan un aspecto oral y un aspecto escrito: la variación ortográfica debe considerarse relevante para la neología.

Basándose en estos cinco postulados, Guilbert (1973) propone cinco tipos principales de neología en la lengua general, que agrupa de la siguiente manera:

- Neología fonológica.
- Neología sintáctica.
- **Neología semántica**.
  - Cambio del agrupamiento de los semas aferentes a un lexema.
  - Cambio de categoría gramatical de un lexema.
  - Sociológico.

- Préstamo.
- Neología gráfica.

La neología fonológica se entiende como un fenómeno que consiste en la creación de secuencias fónicas inéditas con un significado novedoso. Por su parte, la neología sintáctica incluye toda formación creada a partir de la combinación de elementos preexistentes en la lengua. Por ejemplo, la combinación se presenta en un aspecto léxico (base y afijo) más un aspecto fraseológico. Otras formas incluyen la composición, derivación sintáctica, locuciones (verbales, adverbiales, preposicionales) y la siglación.

En cuanto a la neología semántica, es definida como todo cambio de sentido que se produce en uno de los tres aspectos principales del lexema sin que ocurra un cambio en la forma del significante de dicho lexema. Esto puede ocurrir de tres formas: la primera en el cambio de grupos de semas aferentes a un lexema, según las diversas modalidades; la segunda afecta a la categoría gramatical del lexema, llamada neología por conversión; y la tercera, denominada neología sociológica, se da por el paso de términos especializados a la lengua general.

Por último, dos tipos de neología corresponden a los préstamos y la neología gráfica. El primer tipo, por préstamo, consiste en el paso de un signo lingüístico desde una lengua de origen —donde funciona según las reglas propias del código de dicha lengua— hacia una lengua distinta donde se inserta en un nuevo sistema lingüístico. Y el segundo, la neología gráfica, proviene de la oposición entre la lengua oral y la escrita, es decir, el paso de un código al otro permite la creación de nuevas formas.

Posteriormente, Moeschler (1974) diferencia entre neologismos *ordinarios* y NS: en el primer tipo agrupa aquellas unidades que tienen una forma y un significado nuevos, mientras que los NS son definidos como unidades preexistentes que adquieren un nuevo significado. Estos últimos son descritos como un caso especial de polisemia, puesto que pueden adquirir nuevos significados a través del tiempo. Además, pueden ser identificados por su contexto de uso, por el contexto de la frase o del sintagma donde aparecen y por el dominio discursivo de referencia. Así, para Moeschler, estas características permiten identificar las palabras que adquieren un nuevo significado, y su carácter polisémico se define en función de la variación y del uso en la lengua. La neología semántica se considera el producto de un cambio simultáneo en tres aspectos: en la combinatoria de la unidad, en el referente que ha sido creado o modificado por dicha combinatoria y en la relación que existe entre significado y referente —creada mediante metáforas, metonimias o juegos de palabras— (Moeschler, 1974, p. 19).

A mediados de los años setenta, la tipología presentada por Guilbert (1975) considera los NS como un tipo de creación neológica que abarca tanto el cambio de significado como el cambio de categoría gramatical. Dicha tipología incluye las siguientes cuatro clases de neologismos:

- Neología fonológica.
- **Neología semántica**.
- Neología por préstamo.
- Neología sintagmática.

Desde una perspectiva saussureana, los NS son el resultado de un proceso mediante el cual un elemento fonológico preexistente adquiere un nuevo significado. En otras palabras, el significante —que sirve como base y existe previamente dentro del léxico— no sufre ningún tipo de modificación morfo–fonológica ni intra–lexemática, de forma que constituye una nueva unidad de significación (Guilbert, 1975, p. 64). Guilbert explica también las dicotomías *monosemia–polisemia* y *polisemia–homonimia* como parte del marco referencial para comprender la neología semántica. En ambos casos, las figuras retóricas (o procesos de mutación semántica) —como la sinécdoque, la metáfora y la metonimia— propician la creación de nuevos significados.

A la par, la tipología de Goose (1975) entiende los NS según la propuesta de Guilbert (1971), por lo tanto, no proporciona definiciones de cada tipo de neologismo, sino que presenta descripciones y ejemplos de cada proceso de creación neológica. La tipología de este autor se muestra a continuación:

- Derivación.
  - Sufijos verbales.
  - Sufijos nominales "abstractos" que indican la acción y el resultado.
  - Sufijos nominales "concretos" que designan a los "actores".
  - Sufijos adjetivales.
- Composición.
- Préstamo.
- Otros procesos.
  - Palabras nuevas.
  - Abrebiaciones.
  - Usos nuevos.
  - **Significados nuevos**.

Los procesos que permiten que ocurra un cambio de significado en una unidad ya existente dentro del repertorio léxico de una lengua son la metáfora (denominativa y estilística), la metonimia y el cambio de campo de aplicación; además, existen otros tres procesos más que pueden dar origen a un nuevo significado (Goose, 1975):

- Por préstamos de lenguas extranjeras que se introducen en la lengua.
- Por adquisición o pérdida de un carácter peyorativo.
- Por una interpretación errónea del significado de una palabra.

Por su parte, Picoche (1977) apunta que situaciones como la invención de un objeto nuevo, la introducción de un producto en el mercado o la definición de un concepto novedoso implican una reestructuración del inventario léxico, que puede producirse mediante especialización o extensión de significado de una palabra existente, por préstamo o por

formación de una nueva unidad. Para esta autora, las creaciones neológicas surgen a partir de las nuevas realidades que la lengua necesita representar y, generalmente, van de la mano del trabajo terminológico. Teniendo en cuenta estos supuestos, Picoche (1977, pp. 134–135) ofrece una breve clasificación de los tipos de neologismos que considera más frecuentes:

- **Neologismos por el desarrollo de un nuevo significado de una palabra antigua**.
- Neologismos por derivación a partir de un nombre común.
- Neologismos por abreviación de una palabra científica.
- Por lexicalización de una sigla y derivación a partir de la misma.
- Por préstamos de una lengua extranjera.

Los NS (*Neologismos por el desarrollo de un nuevo significado de una palabra antigua*) se ejemplifican mediante una serie de palabras. Por ejemplo, *forchette* ("tenedor"), asociado habitualmente con la acepción culinaria, adquiere un nuevo significado relativo a la probabilidad de que ocurran dos eventos; *couverture* ("cobertura") suma un nuevo significado en el campo jurídico-militar; *sanctuaire* ("santuario"), además de ser un templo religioso, denomina un territorio de carácter militar. Mediante estos ejemplos —y los proporcionados para los otros tipos de neologismos— el lector debe inferir la definición de NS, ya que no se proporciona ninguna más allá de su nombre.

A principios de la década de los ochenta, Leduc-Adine (1980) realiza un estudio sobre la producción terminológica y la producción neológica, y la vinculación entre ellas, ya que, a su juicio, ambos procesos ocurren en un nivel léxico y denominativo. En el caso de la neología, los cambios que se producen no están relacionados con otros dominios de la lengua, como la fonología o la sintaxis. Partiendo de esta premisa, el autor presenta la siguiente clasificación de neologismos terminológicos:

- Neologismos formales.
  - Por el sistema derivacional.
  - Por el sistema de composición.
  - Neología sintagmática.
- **Neologismos semánticos**.
- Neologismos por préstamo.

El NS se refiere a la adquisición de un nuevo significado por parte de un significante preexistente, de tal forma que se crea una nueva unión entre un significante y un significado. El nuevo significado puede ser identificado en la lengua general (*macro–contexte*) por la estructura y el tipo de texto, y por el significado especializado de la unidad (*micro–contexte*) que el NS denota dentro de la lengua general[1].

[1]Bastuji (1974) indica que la neología semántica se produce o detecta por su contexto de aparición: el microcontexto de la frase o sintagma donde se usa el neologismo y el macrocontexto del dominio discursivo de referencia al que pertenece el texto.

Cuatro años más tarde, Walter (1984) realiza un estudio donde analiza el habla y el léxico de los jóvenes parisinos con la finalidad de documentar el uso y creación de palabras nuevas en francés. A partir de los resultados de esta investigación, Walter establece la siguiente clasificación de neologismos:

- Nuevos significados.
  - **Nuevos significados por polisemia simple**.
  - Nuevo significado manifestado por un cambio de construcción sintáctica.
  - Nuevos significado por transferencia de categoría.
    - Sin modificación del significante.
    - Con modificación del significante.
- Nuevos significantes.
  - Nuevos significantes por derivación.
    - Sufijos.
    - Prefijos.
  - Nuevos significantes por abreviación.
  - Nuevos significantes por préstamo.
    - Préstamos de lenguas extranjeras.
    - Préstamos argóricos.
  - Nuevos significantes por onomatopeyas.
  - Casos particulares de expresiones negativas.

De acuerdo con esta categorización, los NS forman parte del grupo de nuevos significados creados por polisemia simple (*Nouveau signifié par polysémie simple*) que Walter (1984) define como la adición de un nuevo significado a un término ya existente, generalmente por un uso metafórico. El hecho de que el significante no altere su forma permite al interlocutor tener cierta conciencia sobre el significado novedoso (Walter, 1984, p. 72). Cabe destacar que si bien el autor indica que este es el tipo de neologismos registrados con mayor frecuencia, en estudios posteriores (ver sección 3.2) se ha documentado que esta clase de neologismos son menos frecuentes en los corpus textuales.

A mediados de los ochenta, en una investigación realizada por Deloffre (1985) sobre el léxico de la obra de Jean-Jacques Rousseau, se exploran los diferentes tipos de neologismos desde una perspectiva histórica, puesto que resulta difícil establecer en qué medida una palabra y especialmente un significado son nuevos o neológicos en una época determinada (Deloffre, 1985, p. 25). Por consiguiente, este estudio no plantea una teoría que explique los neologismos, sino que muestra la influencia que puede tener un agente externo (en este caso, un autor, Rousseau) sobre una lengua. De los resultados de la investigación se deriva la siguiente tipología de neologismos:

- Palabras nuevas préstamos de lenguas extranjeras.
- Palabras compuestas por derivación propia.

- Palabras compuestas por derivación impropia.
- Alargamiento de las construcciones verbales.
- **Diversas extensiones de significado o de uso**.

Los NS pertenecen a la categoría denominada "diversas extensiones de significado o de uso" (*diverses extensions de sens ou d'emploi*, en francés). Si bien no se aporta una definición formal sobre los NS, sí se enumeran los distintos tipos de extensiones que pueden propiciar un cambio de significado, a saber: "passage du sens propre au sens figuré, du sens technique au sens général, de l'emploi défavorable à l'emploi favorable et réciproquement, ou même simplement mots à la mode dans telle ou telle acception" (Deloffre, 1985, p. 26). Del listado anterior podemos destacar que el paso de un significado técnico a uno general también constituye un tipo de neologismo semántico.

Siete años después, en un estudio realizado por Giardina (1992) rsobre la obra de Boris Vian, *L'Ecume des jours*, se destaca el papel de la creación léxica dentro de este libro. Giardina (1992) detecta los seis procesos siguientes de creación neológica, que se limita a ejemplificar:

- Creación de una palabra nueva a partir de una palabra preexistente.
- Formación de palabras compuestas.
- Palabras preexistentes empleadas con un significado no habitual.
- **Palabras basadas en un cambio semántico o de categoría gramatical**.
- Palabras que tienen una grafía nueva.
- Contaminación: concatenación de dos palabras preexistentes para formar una palabra nueva.
- Otras creaciones léxicas.

Esta tipología no cuenta con una única categoría que comprenda los NS, sino que los ejemplos de esta clase de neologismos se encuentran distribuidos entre los procesos "palabras preexistentes que se emplean con un significado no habitual" y "palabras basadas en un cambio de categoría gramatical o semántico" (en francés, (*mots qui existent employés dans un sens inhabituel* y *mots basés sur un changement de catégorie grammaticale ou sémantique*, respectivamente). De este modo, dentro de ambas categorías se describen procesos de cambio semántico.

Por una parte, el primer proceso engloba las palabras que tienen cierta similitud formal con otras de significado diferente y que se usan con un significado nuevo para crear un efecto humorístico. Por otra parte, la segunda categoría incluye ejemplos de cambio semántico mediante figuras retóricas[2], como es el caso de *pandour*, que primeramente describe a un soldado o a una persona tosca y que, por extensión, se usa también para hablar de un tipo de piel similar al cuero o la gamuza en el contexto de *peau de pandour*

[2]El autor no presenta explícitamente el uso de figuras retóricas, pero la lectura de los ejemplos que proporciona nos lleva a inferir el juego entre la parte por el todo y metáforas.

("piel de pandour"). En definitiva, Giardina (1992) lleva a cabo una descripción detallada de los noventa neologismos detectados en la novela *L'Ecume des jours*, pero la tipología y el trabajo de clasificación no presentan los NS como una categoría independiente.

Por último, la tipología de Rey (1995) clasifica la neología en tres grupos principales: a) formal, b) semántica y c) pragmática. El primero corresponde a los neologismos que se forman mediante las reglas de sufijación, composición y prefijación del sistema de la lengua en el que se introducen. La novedad semántica puede de los neologismos formales puede ser 1) total dentro del sistema, como es el caso de los préstamos lingüísticos; 2) parcial, como sucede en los elementos creados a partir de afijación, composición, aglutinación en palabras complejas, o formaciones en grupos de palabras; o 3) muy débil, como en los acrónimos y las abreviaciones, que solo expresan la forma a la cual hacen referencia, pero al tratarse de una forma compacta, cambian su denotación.

El segundo grupo, relativo a los NS, tiene en cuenta que este tipo de neología constituye un fenómeno presente de forma general en todos los neologismos[3]. Rey (1995). El tercer grupo se encuentra definido por la función comunicativa, ya que para Rey es imposible concebir un neologismo de manera abstracta, es decir, como un elemento nuevo dentro de un sistema que funciona de forma independiente a otros procesos concretos del lenguaje. La percepción de novedad no puede ser independiente de los procesos internos de la lengua, puesto que una unidad puede ser considerada neológica según la percepción de cada uno de los interlocutores.

3.2 Dos posturas contemporáneas sobre la clasificación de neologismos

El primer trabajo de Sablayrolles (1996), relacionado con una propuesta teórica sobre la neología y los neologismos, se fundamenta en el concepto de las lexías neológicas. Estas lexías tienen funcionalmente el mismo estatus y distribución que las palabras. Semánticamente, son unidades que tienen estabilidad referencial o permiten una estabilidad que antes no existía, y también apelan a la memoria de dos formas: por una parte, dan cuenta de lo existente y, por otra, interpretan lo nuevo en función de lo ya conocido.

Como continuación de este primer planteamiento, Sablayrolles (2000) retoma la concepción de los neologismos como lexías neológicas. Asimismo, se basa en la clasificación de neologismos de Tournier (1985, 1991) —que fue originalmente diseñada para esquematizar y encapsular los diferentes procesos de creación neológica en inglés— y la modifica para adaptar los mecanismos de innovación al francés y a su propuesta teórica.

Dentro de esta tipología, los NS se conciben como palabras que adquieren un nuevo significado sin que cambie su significante (Sablayrolles, 2000, p. 226). Los métodos que favorecen la creación de esta clase de neologismos son mayoritariamente figuras retóricas, de las cuales la metonimia y la metáfora constituyen los procedimientos más productivos. La lista que se muestra a continuación presenta los distintos métodos que permiten la formación de NS (Sablayrolles, 2000, p. 245):

- Extensión de significado, empobrecimiento de significado.

[3]Dentro de esta tipología Rey (2005) define los NS como unidades lingüísticas que nombran una nueva relación de sentido, sin importar que el concepto designado sea innovador o no.

- Restricción de significado, enriquecimiento de significado.
- Remotivación, reactivación, reactualización.
- Etimología popular, falsa etimología.
- Metáfora.
- Metonimia.
- Sinécdoque.
- Antonomasia.
- Atenuación.
- Antifrase.
- Oxímoron.
- Hipocorística, términos de cariño.
- Calambur, juego fónico/gráfico.
- Anfibología.
- Paradoja.

En esta tipología de neologismos[4], Sablayrolles (2000) incluye modificaciones del trabajo de Tournier (1985) y reestructuraciones en las matrices internas en los siguientes niveles: sufijación, procesos de construcción por afijación, composición, juego fonético, juego semántico, cambio de significado y reducción de significado. En particular, en el nivel de cambio de significado, agrupa los procesos de metáfora, metonimia, otras figuras retóricas, y restricciones o extensiones de uso.

Esta primera versión se revisa posteriormente Sablayrolles (2006) y da como resultado la tipología que se muestra en la tabla 3.1, donde se puede observar que los diversos procesos de creación de lexías neológicas se encuentran agrupados por afinidades y tienen como ejes dos matrices principales: la matriz externa, que corresponde a los préstamos, y la matriz externa, que incluye procesos morfo–semánticos, sintáctico–semánticos, morfológicos y pragmáticos.

Como se puede ver en la tabla, los NS son un proceso de matriz interna que ocurre en el nivel sintáctico–semántico y que pueden desarrollarse principalmente a través de la metáfora, la metonimia u otras figuras y procesos que permiten la creación de nuevos significados. Dentro de este nivel, concretamente en la categoría *Autres* ("Otros"), Sablayrolles (2006) introduce una modificación con respecto a la tipología original añadiendo la creación de nuevos significados por eufemismo.

En definitiva, la propuesta de Sablayrolles (2006) es un marco teórico que intenta dar explicación al fenómeno de la neología planteando una clasificación que agrupa los diferentes procesos que favorecen la innovación léxica dentro de cinco categorías principales. No obstante, Cabré (2009, p. 34) formula una serie de críticas hacia este acercamiento:

[4]También llamada *matriz lexicogénica* según los supuestos teóricos de Tournier (1985), que son expandidos en Tournier (1991).

<table>
<tr><td rowspan="12">Matrices internes</td><td rowspan="9">Morpho-sémantiques</td><td rowspan="5">Construction</td><td rowspan="4">Affixation</td><td>Préfixation</td></tr>
<tr><td>Suffixation</td></tr>
<tr><td>Dérivation inverse</td></tr>
<tr><td>Parasynthétique</td></tr>
<tr><td colspan="2">Flexion</td></tr>
<tr><td rowspan="2">Composition</td><td>Composition
Synapsie
Quasimorphème</td></tr>
<tr><td>Mot valise</td></tr>
<tr><td colspan="2">Imitation et déformation</td><td>Onomatopée
Fausse coupe
Jeu graphique
Paronymie</td></tr>
<tr><td rowspan="5">Syntactico-sémantiques</td><td colspan="2" rowspan="2">Changement de fonction</td><td>Conversion</td></tr>
<tr><td>Combinatoire
Syntaxique/Lexicale</td></tr>
<tr><td colspan="2" rowspan="3">Changement de sens</td><td>Métaphore</td></tr>
<tr><td>Métonymie</td></tr>
<tr><td>Autres</td></tr>
<tr><td rowspan="2">Morphologiques</td><td colspan="2" rowspan="2">Réduction de la forme</td><td>Troncation</td></tr>
<tr><td>Siglaison</td></tr>
<tr><td colspan="3">Pragmatique</td><td>Détournement</td></tr>
<tr><td colspan="4">Matrice externe</td><td>Emprunt</td></tr>
</table>

Tabla 3.1 – Matriz lexicogénica (Sablayrolles, 2006, p. 146).

- No veiem clara la distinció entre processos morfosemàntics, sintacticosemàntics i purament morfològics atès que no s'adecua a la nostra concepció de la formació de paraules en el marc d'una gramàtica. En la nostra opinió, seria preferible partir dels tipus d'intervenció gramatical associats al tipus de procés que du a terme[...]

- No acabem de veure pertinent la distinta ubicació de la conversió i la derivació inversa, ja que semblen ser mateix procés.

- No considerem que les formacions sintagmàtiques s'incloguin en la composició si es un concep aquest terme en sentir estricte com a tipus de combinació, i tampoc s'ajusta a les nostres consideracions gramaticals.

- No considerem la formació d'una sigla corresponent a un nom propi com a neologisme lèxic pròpiament dit, etc.

De estas consideraciones se desprende que la tipología de Cabré (2009) —cuya propuesta actualizada se muestra en la tabla 3.2— surge para dar respuesta a las necesidades del proyecto NEOROM y específicamente a las del Observatori de Neologia de la Universitat Pompeu Fabra, que no se ven cubiertas en su totalidad por la clasificación de Sablayrolles (1996, 2000, 2006), ya que el trabajo que se lleva a cabo en los observatorios de neología consiste en la aplicación de una serie de filtros y criterios que deben seguirse sistemáticamente para llevar a cabo el vaciado de neologismos (Cabré y Estopà, 2004b; Cabré et al., 2004; Estopà, 2009; Freixa, 2009).

Esta primera tipología (Cabré, 2009) tiene la finalidad de homogeneizar los datos recopilados por los equipos de trabajo y, así, facilitar su análisis, pero se considera un primer

paso que requiere revisión. La subsecuente revisión (Cabré, 2011b) da como resultado la clasificación presentada en la tabla 3.2, que establece cinco tipos principales de neologismos partiendo de su carácter poliédrico: formales, sintácticos, semánticos, préstamos, y otros, en los que se incorporan fenómenos menos frecuentes.

Variación	Gráfica				
	Fonológica (en los casos que no se trate de una variante ortográfica)				
	Creación			Sí	
				No	
	Formación	Combinación	Combinación morfológica	Prefijación	Prefijo actual
					Prefijo grecolatino
					Prefijoide
				Sufijación	Sufijo actual
					Sufijo grecolatino
				Prefijación y sufijación	
				Parasíntesis	
				Composición	Composición patrimonial
					Composición culta
					Composición híbrida
			Combinación sintáctica (especificar el núcleo sigla/unidad léxica)		
			Repetición		
		Cambio	Cambio gramatical	Cambio de categoría gramatical	
				Cambio de subcategorización	
			Resemantización	Reducción de significado	
				Ampliación de significado	
				Cambio de significado	
			Reducción	Siglación	
				Acronimia	
				Abreviación	
			Fijación o lexicalización de una forma flexiva		
	Préstamo	Origen lingüístico: especificar lengua			
		Procedencia de la lengua	Del mismo alfabeto		
			De distinto alfabeto	Transcripción	
				Transliteración	
				Mixto	
		Préstamo directo/préstamo a través de otra lengua			
		Adaptación a la lengua de acogida	Sí		
			No		
		Tipo de adaptación		Gráfica	
				Fónica	
				Morfológica	
Estructura interna	Simple				
	Construida (reproducción de la estructura jerárquica)				
Agente neológico	Planificado				
	Espontáneo				

Tabla 3.2 – Clasificación multivariante de neologismos (Cabré, 2011b, p. 485).

Esta tipología no es una propuesta teórica sobre los neologismos, ya que "no existe una teoría morfológica precisa, sino un conjunto de fundamentos que sin contradicción entre sí justifican cada una de las elecciones que se han hecho" (Cabré, 2011b, p. 486). No obstante, la motivación y el propósito de esta clasificación se adecuan a la finalidad y objetivos de nuestra tesis, puesto que permite delimitar y categorizar sistemáticamente a los NS. Para nuestra aplicación resulta más operativo adecuarnos a una propuesta que ha permitido sistematizar el vaciado de neologismos, ya que servirá de referente para el diseño del flujo de trabajo de la aplicación y para delimitar conceptualmente el tipo de unidades que se espera extraer.

Dentro de este enfoque, el NS se define como un neologismo formado por una modificación del significado de la base léxica, como se ve en los ejemplos (1) y (4), relativos a las palabras *navegador* y *tableta*[5]. La modificación conlleva una reducción, ampliación o cambio de significado que puede producirse por medio de distintos procesos, como el paso de un nombre propio a un nombre común y el uso de figuras retóricas, principalmente la metáfora y la metonimia[6].

1. La eclosión de Internet pilló a Microsoft, por sorpresa, pero reaccionó con rapidez. Ideó el Explorer, un *navegador* para internet que mejora notablemente con cada versión, pero que no tiene la buena prensa de su competidor Netscape.

2. Entre los alumnos también hay un rechazo al uso tecnológico. Utilizan el *messenger*, los foros, pero a la hora de publicar en un blog o trabajar sienten más pereza.

3. Algunas de las perlitas que se han podido leer en la *wikipedia* es que Hitler fue un filántropo humanista o que los suecos descubrieron América.

4. Un poco antes, 2013, será el año de las *tabletas*: el 80 % de las empresas confiará en ellas de una forma u otra.

De acuerdo con esta metodología de trabajo, se debe señalar si una palabra es candidata a ser considerada neológica con respecto a una de las acepciones (recogidas en los diccionarios de referencia) o con respecto a todas ellas (Cabré y Estopà, 2004b). En los ejemplos (1) y (4), *navegador* y *tableta* representan un nuevo significado en relación con todas las acepciones y, en la actualidad, ya se encuentran registradas en el diccionario como parte de la lengua general. Los NS también pueden formarse a partir de nombres propios, que pasan a ser utilizados como nombres comunes, como es el caso de los ejemplos (2) y (3), correspondientes a las palabras *messenger* y *wikipedia*.

La detección de neologismos semánticos es una tarea más difícil que la detección de neologismos formales. Sin embargo, dentro de este fenómeno hay casos más claros, como los mencionados en los ejemplos anteriores, y otros más complejos, como aquellos que se dan por ampliación o reducción de significado, aquellos cuyo significado nuevo no se aleja del significado estricto de la unidad o aquellos cuyo significado argótico acaba incorporándose en el lenguaje coloquial (Cabré y Estopà, 2004b).

[5]Los ejemplos fueron extraídos de la base de datos del OBNEO.

[6]Una descripción en profundidad de estos procedimientos en el marco de esta metodología de trabajo se encuentra en Feliu et al. (2009).

Además de esta clasificación, la metodología de trabajo descrita por Cabré et al. (2004) aporta una serie de filtros para determinar el grado de neologicidad de un candidato a neologismo. Entre los filtros generales se mencionan los siguientes: la presencia de los neologismos en otras fuentes de referencia, la frecuencia de aparición en la base de datos del OBNEO y la presencia de cambios menores respecto a variantes ortográficas registradas anteriormente siguiendo el criterio lexicográfico. En particular, para los NS se añaden los siguientes criterios (Cabré et al., 2004, p. 239):

- La utilització de noms propis com a noms comuns dóna com a resultat unitats perceptivament molt neològiques, és a dir amb un grau de neologicitat alt.
- El canvi radical de significat en relació a la unitat documentada en el corpus lexicogràfic d'exclusió també comporta un grau de neologicitat major que si només es tracta d'una ampliació o restricció de l'acepció.
- El salt d'un àmbit temàtic a un altre es percep com un grau de neologicitat major que si el neologisme semàntic i la unitat documentada respecte a la qual és neològica pertanyen al mateix àmbit temàtic.

El tercer criterio resulta de particular interés, ya que puede dar pie a la sistematización de un proceso de detección de NS. Como supuesto de trabajo en esta tesis, el cambio de temática puede tratarse mediante un modelo de clasificación de documentos, donde cada categoría o clase del modelo representa una temática de un campo especializado que puede utilizarse para comparar dicha temática frente a un modelo de lengua general. La discordancia entre la temática existente en la lengua general y la especializada puede determinar si, dentro de un texto, una palabra puede ser un candidato a NS. En conclusión, el fenómeno de la NS se tratará desde la perspectiva del marco metodológico del OBNEO (Cabré y Estopà, 2004b,a; Cabré et al., 2004), la tipología propuesta por Cabré (2011b) y los trabajos que surgen a partir de dichos enfoques.

3.3 Las unidades terminológicas dentro de la Teoría Comunicativa de la Terminología

Para el desarrollo del sistema de detección y extracción de neologismos semánticos que proponemos en esta investigación, adoptamos el principio de la adecuación —propio de la Teoría Comunicativa de la Terminología (TCT) (Cabré, 1999)— como medida de delimitación y definición de los términos procedentes de la informática que se insertan en la lengua general. El concepto de *término* ha sido definido desde diferentes escuelas de pensamiento, cada una de las cuales ha propuesto una visión distinta en función de sus necesidades, supuestos teóricos o acercamiento metodológico. Desde la Teoría General de la Terminología (TGT) (Wüster, 1973, 1979, 1998), que constituye el primer referente en este ámbito, los términos son concebidos como unidades que definen conceptos y cuya finalidad es normalizar el conocimiento científico–técnico, circunscritos al ámbito de uso especializado. Esta ha sido criticada por muchos especialistas, pues tiende a reducir el potencial lingüístico del término al dar por supuesto que el conocimiento terminológico solamente es usado por especialistas en un contexto de especialidad. Posteriormente, se

han desarrollado otras cuatro corrientes teóricas sobre la terminología, que, según Cabré et al. (2018), pueden agruparse en torno a dos perspectivas:

- Teorías y perspectivas sociales y comunicativas:
  - La socioterminología (Boulanger, 1991; Guespin, 1991; Gaudin, 1993; Faulstich, 1995).
  - La teoría comunicativa de la terminología (TCT), representada por la escuela de Barcelona (Cabré, 1992, 1999, 2011a)[7].
- Aproximaciones basadas en una orientación cognitiva:
  - La terminología sociocognitiva (Temmerman, 1997, 2000, 2001).
  - La terminología basada en marcos (FBT) (Faber, 2012, 2015).

Nuestro estudio se enmarca dentro de las perspectivas sociales y comunicativas, puesto que hemos optado por ceñirnos a los principios teóricos y metodológicos de la TCT, puesto que considera que el uso del conocimiento terminológico no se encuentra limitado a los contextos de especialización, sino que el contexto comunicativo donde son empleados —así como sus usuarios— pueden tener diferentes grados de especialización. Dentro de esta visión, *término* y *unidad terminológica* (UT) se consideran variantes sinónimas (Cabré, 2011a) y se definen, en primera instancia, de la siguiente manera:

> Los términos son unidades léxicas, activadas singularmente por ser condiciones pragmáticas de adecuación a un tipo de comunicación, que se componen de forma o denominación y significado o contenido. La forma es constante; pero el contenido se singulariza en forma de selección de rasgos adecuados a cada tipo de situación y determinados por el ámbito, el tema, la perspectiva de abordaje del tema, el tipo de texto, el emisor, el destinatario y la situación. (Cabré, 1999, p. 123)

Esta noción de *término* se complementa con una definición posterior que contempla las condiciones que una unidad léxica deben de cumplir para ser considerada UT: a) estructura, b) especificación y c) necesidad de la estructura conceptual. Estas condiciones se cumplen cuando la estructura de una unidad léxica "corresponde a una unidad léxica de origen o producto de la lexicalización de un sintagma, que posee un significado específico en el ámbito al que se asocia y es necesaria en la estructura conceptual del dominio del que forma parte" (Cabré y Estopà, 2005, p. 77).

El valor terminológico de las UTs se encuentra definido por las condiciones pragmáticas de un tipo de comunicación determinada y, al mismo tiempo, las UTs poseen un significado específico dentro de un ámbito especializado. Las UTs también se componen por denominación y contenido, donde dicho contenido se delimita en función de la temática y la situación comunicativa donde la UT se utiliza. En consecuencia, se podría concebir la noción de UT como una especie de contenedor que puede dar cuenta de una

[7]Se menciona el trabajo de Cabré (1992) ya que puede ser considerado un primer acercamiento a una propuesta teórica sobre la terminología. No obstante, en un sentido estricto, esta obra no se enmarca por completo dentro de los principios de la TCT que son desarrollados en las obras posteriores de la autora.

realidad distinta manteniendo su forma, de modo que cada realidad representada corresponde a una faceta distinta de una misma UT. Esta propuesta se denomina “principio del valor terminológico"(Cabré, 2011a, p. 6).

Este principio propone una delimitación clave para nuestro estudio: no solamente la temática por sí misma determina la pertenencia de una UT a un campo del conocimiento, sino que la totalidad del léxico especializado sirve para crear el constructo conceptual que activa el significado de las UTs. Podríamos pensar en nodos dentro de una red que, en conjunto, activan el valor especializado de una UT, que, al mismo tiempo, forma parte de este repertorio y también cuenta con la capacidad de ser un nodo activador. También podríamos señalar que, como contraparte, el inventario léxico de la lengua general y su interrelación serían un elemento clave de contraste para delimitar las UTs.

Aparte de la noción de UT, podemos destacar el principio de la adecuación, uno de los principios fundamentales de la TCT. De acuerdo con este principio, “cada trabajo en concreto se adapta a una estrategia en función de su temática, objetivos, contexto, elementos implicados y recursos disponibles[...]. La adecuación puede adoptar una perspectiva onomasiológica o semasiológica; puede partir de textos o de bancos de datos; puede procesar automáticamente textos en soporte digitalizado y aplicar detectores semiautomáticos que exigirán una profunda labor de revisión” (Cabré, 1999, p. 137).

Esta propuesta teórico–metodológica surge para dar explicación a casos problemáticos (como la detección de UTs polisémicas o UTs sinonímicas) que no se encuentran recogidos dentro del marco de teorías clásicas como la TGT. La TCT nos permite tener en cuenta la dimensión social y la comunicativa, y contrastar UT formalmente idénticas que pertenecen a diferentes dominios, puesto que estas unidades activan su valor terminológico en función del repertorio léxico del campo al que pertenecen (Cabré, 1999, p. 147).

3.4 Unidades terminológicas e innovación léxica

El carácter neológico de un término pasa por dos momentos claves: en primer lugar, aparece inicialmente en su campo de especialización y en segundo lugar, se introduce en la lengua general. Como se ha mencionado en el apartado anterior, consideramos que la dirección del cambio de significado se produce desde lo especializado hacia lo general. Guerrero Ramos (2015) parte de las ideas de Rey (1976) y Rondeau (1984) para definir los conceptos de *neonimia* y *neónimo*. Los neónimos son un tipo de neologismos que surgen cuando un término nuevo aparece en una lengua de especialidad (o lenguaje especializado) en el momento en el que una nueva noción o realidad es creada. Este tipo de innovación léxica es similar a los fenómenos observados por Guilbert (1971) y Deloffre (1985)[8], quienes observaron que los campos de la ciencia y la tecnología son focos de innovación léxica terminológica.

El uso de neónimos no está restringido a contextos de elevada especialidad, sino que estas unidades pueden encontrarse también “en textos de divulgación, banalizados, como puede ser la prensa especializada e, incluso, la general” (Guerrero-Ramos y Pérez-Lagos, 2012, p. 27). De acuerdo con Adelstein (1996), el proceso de banalización de un término depende de su difusión en el entorno social, ya que cuando “el dominio especializado

[8] Ambas posturas se describen en la sección 3.1.

constituye un sector esencial de la cultura o de la vida de la sociedad, las términos tienen todas las posibilidades de penetrar al léxico común sin restricciones" (Adelstein, 1996, p. 41).

El uso de neónimos no se encuentra acotado al contexto de la especialidad, Guerrero-Ramos y Pérez-Lagos (2012) menciona que estas unidades pueden encontrarse "[...]no solo en los discursos altamente especializados escritos por especialistas y cuyos receptores son también especialistas, sino que se pueden encontrar en textos de divulgación, banalizados, como puede ser la prensa especializada e, incluso, la general." (p.27). De acuerdo con Adelstein (1996) el proceso de *banalización* de un término depende de la difusión que dicho concepto tiene en el entorno social y especifica que cuando "el dominio especializado constituye un sector esencial de la cultura o de la vida de la sociedad, las términos tienen todas las posibilidades de penetrar al léxico común sin restricciones".

Por consiguiente, los NS analizados en esta tesis no solo han sido producto de un proceso de cambio semántico, sino que también, dentro del ámbito especializado, han denominado una nueva realidad o concepto y, en la lengua general, dicho concepto ha sufrido un proceso de banalización[9]. Partimos del supuesto de que el dominio de la informática no solo es productivo terminológicamente, sino que, gracias a la implantación de las tecnologías de la información en nuestra vida cotidiana, la difusión de dicho conocimiento permite la incorporación de UTs procedentes de este ámbito en la lengua general.

A pesar de que la informática es una categoría que empíricamente es considerada rica en producción terminológica, Guerrero-Ramos y Pérez-Lagos (2012) señalan que los NS son generalmente una categoría improductiva: del total de 5137 neologismos recogidos desde 2004 hasta 2010, se documentaron 179 NS y, de ese subconjunto, solo hay registro de 2 NS especializados. En los capítulos 5 y 6 de esta tesis, podemos observar esta misma tendencia con el conjunto de neologismos del OBNEO y el subconjunto de NS informáticos que fue generado para este estudio.

En conclusión, en nuestra investigación los principios de la TCT nos permiten realizar la detección de nuevos significados banalizados en lengua general, en concreto, nuestro sistema detecta aquellas unidades léxicas preexistentes que extienden un nuevo significado terminológico al discurso general. En otras palabras, tratamos de detectar UTs que aparecen en una situación comunicativa distinta, donde su significado terminológico adquiere un carácter neológico en contraste con el significado habitual de una unidad léxica formalmente idéntica.

[9]Cabe señalar que la banalización es en sí misma un proceso de creación neológica y no el proceso de difusión de un término.

Capítulo 4

METODOLOGÍA

Como se ha mencionado en los capítulos 2 y 3, las características de los neologismos semánticos (NS), complican la tarea de detección semiautomática y automática y, por lo tanto, también el diseño de una herramienta para su detección. La mayoría de los programas de detección de neologismos no contemplan a los NS, puesto que su metodología de trabajo principal suele ser el criterio lexicográfico. Este criterio depende del uso de listas de exclusión, diccionarios y reglas morfológicas para determinar si una palabra es un candidato válido a neologismo.

El criterio lexicográfico no permite diferenciar o detectar unidades como los NS puesto que se limita a realizar un análisis formal, es decir, no toma en cuenta la posibilidad de que una palabra conserve su forma y adquiera un nuevo significado. Para ilustrar esta limitación tomamos las entradas *ratón* y *navegador* del *Diccionario de lengua española* de la Real Academia Española (en adelante DLE), ya que la segunda acepción de ambas entradas recoge un significado que fue considerado neológico en un momento histórico previo a su incorporación en el DLE:

> DLE:
> ratón, na
> De rato.
> 1. m. y f. Mamífero roedor de pequeño tamaño, de hocico puntiagudo y cola larga, de pelaje generalmente gris, muy fecundo y que habita en las casas. U. en m. ref. a la especie.
> 2. m. Pequeño aparato manual conectado a una computadora, cuya función es mover el cursor en la pantalla para dar órdenes.

> DLE:
> navegador, ra
> Del lat. navigātor, -ōris.
> 1. adj. Que navega. U. t. c. s.
> 2. m. Inform. Aplicación que, mediante enlaces de hipertexto, permite navegar por una red informática.

Palabras como *navegador* y *ratón* han sufrido un proceso de cambio semántico, ya que eran formas existentes en el repertorio léxico del español que han adquirido un nuevo significado. Podemos observar que, en ambos casos, las unidades conservaron su base

léxica y adquirieron un nuevo significado en un ámbito especializado. Como ejemplo de un proceso de cambio semántico en curso podemos mencionar la palabra *viral*, ya que solo cuenta con una acepción en el DLE, sin embargo, tras consultar el CORPES XXI[1] encontramos concordancias que dan cuenta de una nueva acepción de *viral* ya que no corresponden a la definición del DLE.

En los ejemplos obtenidos del CORPES XXI *viral* es un tipo de contenido como imagen, audio, o vídeo que se propaga masiva y rápidamente, generalmente a través de medios digitales. Estas concordancias se diferencian del significado relativo a los virus (que nos remite al campo de la biología) que se encuentra documentado en la entrada del DLE, puesto que los contextos de uso extraídos del CORPES XXI corresponden a un significado novedoso relacionado con el campo de informática. Esta discordancia de temáticas puede servir como indicador de un proceso de cambio semántico en curso.

> DLE:
> viral
> 1. adj. Perteneciente o relativo a los virus.
>
> CORPES XXI:
> [...] o reciben correos y mensajes instantáneos"(IDC, 2009). No se descarta el uso del móvil para postear en los blogs, o enviar mensajes al Twitter, e inclusive, usos publicitarios y de marketing *viral*.
>
> [...] a la contraparte hasta que acceda a abrirse para la consumación de la venta. Hay diversos estilos: la publicidad es un burdo cabaret, el marketing relacional una mala imitación de un mal matrimonio, y el marketing *viral*, que aparecería años después, no es sino una nueva forma de promiscuidad.
>
> ¿Has pensado que quizá podría tratarse del tráiler de una nueva producción X?, ¿un recurso narrativo premeditado, algo así como un manuscrito hallado?, ¿lo más de lo más en publicidad *viral*, paradójicamente, a través de una cinta VHS?, ¿lo último de J. J. Abrams?
>
> Los Ángeles - El portal de vídeos más popular de internet, YouTube, va camino de convertirse en una industria audiovisual profesionalizada, organizada y rentable con una visión de negocio que va más allá del fenómeno *viral*.

Todos estos ejemplos muestran que un candidato a NS aparece en un contexto cuya temática es distinta a las temáticas ya documentadas en los diccionarios de referencia. Dicho de otra manera, las definiciones del diccionario dan cuenta de los significados más comunes de una palabra y, en consecuencia, también recopilan las temáticas que tienen mayor relación con una palabra. Estas particularidades nos permiten establecer los siguientes supuestos de trabajo:

- Las temáticas (o temas) tratadas en las acepciones de un diccionario son simultáneamente las más recurrentes en un corpus de lengua general y, por lo tanto, estos temas pueden ser agrupables y detectables.

[1] `http://www.rae.es/recursos/banco-de-datos/corpes-xxi`

- Un texto puede tratar diversas temáticas y contener palabras que están vinculadas con cada una de dichas temáticas. Por ejemplo, en las concordancias de *viral* encontramos palabras como *internet*, *YouTube*, *marketing* y *Twitter* que sirven para contextualizar al lector y que indican que *viral* se está usando con un nuevo significado perteneciente a un campo del conocimiento distinto al de su significado prototípico.

- Los NS conservan su forma y no son detectables mediante el criterio lexicográfico; no obstante, son unidades relevantes en el discurso ya que, su carácter novedoso, crea nuevas asociaciones entre palabras que focalizan la atención del lector hacia la unidad neológico.

Tomando estos supuestos como puntos de partida, proponemos el sistema DENISE[2], un sistema que implementa la combinación de las siguientes estrategias metodológicas:

- Detección automática de tema, tratada desde la clasificación automática de textos mediante la implementación de un modelo de regresión logística y una ponderación frecuencia de término – frecuencia inversa de documento (TF-IDF).

- Extracción automática de palabras clave mediante aprendizaje automático no supervisado (Mihalcea y Tarau, 2004) en conjunto con filtros de etiquetas gramaticales.

- Desambiguación de significado empleando *word embeddings* neuronales (Bengio et al., 2003; Mikolov et al., 2013b) con etiquetado gramatical supervisado (Trask et al., 2015) para incrementar la precisión del modelo neuronal.

DENISE es un sistema que pretende detectar el cambio semántico de aquellos términos que se introducen en la lengua general y aportan un nuevo significado, es decir son una nueva acepción de una palabra ya conocida. En esta tesis la detección se limita a términos que pertenecen al campo de la informática y que posteriormente se integran a la lengua general con un nuevo significado.

Este sistema toma como entrada un texto plano que puede ser introducido manualmente por el usuario, o bien puede ser extraído automáticamente desde la web. Posteriormente, DENISE realiza las siguientes operaciones de forma automática o mediante selección manual por el usuario, según se considere necesario[3]:

1. Selección manual o detección automática de lengua.

2. Selección manual o detección automática de tema.

3. Extracción automática de palabras clave.

4. Desambiguación de tema y significado de palabras clave mediante un modelo neuronal con etiquetado gramatical.

5. Generación de lista de candidatos a NS.

[2]Un juego de palabras en tres lenguas: *detector de neologismos semánticos*, en español; *detector de neologismes semàntics*, en catalán y *détecteur de néologismes sémantiques* en francés.

[3]Las operaciones descritas intentan dar solución a los problemas que presenta la detección de NS, en la sección 4.3 se explicarán de forma general cada uno de los procesos que lleva a cabo DENISE.

Los modelos neuronales entrenados con corpus generales pueden ser usados para crear representaciones vectoriales de las asociaciones lingüísticas más comunes que existen entre palabras. En otros términos, el uso de *word embedding* como método de desambiguación de signficiado permite obtener las palabras que tienen mayor similitud semántica con una palabra consultada gracias a que emplean corpus de lengua general etiquetados gramaticalmente.

Esta metodología nos permitirá determinar si hay concordancia entre la temática que ha sido detectada en el texto de entrada y la temática de las representaciones vectoriales de cada palabra clave. De esta forma, si no existe concordancia entre los temas detectados en un texto y las palabras más similares de una palabra clave consultada, dicha palabra clave consultada podría ser un candidato a NS.

En los siguientes apartados se describirán los elementos necesario para el flujo de procesos de DENISE: corpus generales, corpus especializados, listas de términos y bases de datos. Todos estos elementos, dada la naturaleza modular de DENISE, cuentan con paralelos en catalán, español y francés. DENISE también tiene la posibilidad de ser implementado en otras lenguas, siempre que se cuenten con los elementos necesarios para realizar el etiquetado gramatical y entrenamiento neuronal que será descrito de forma general en los apartados siguientes y con mayor profundidad en el capítulo 5.

4.1 Conceptos generales sobre aprendizaje automático supervisado y no supervisado

Apegándonos a los conceptos definidos por Hassoun (1995) podemos decir, a grandes rasgos, que el aprendizaje automático se lleva a cabo mediante un proceso de adaptación, dicho proceso también puede ser definido como *reglas de aprendizaje* o *algoritmo*, donde los pesos de una red se ajustan de forma incremental para mejorar una medida de desempeño predeterminada a través del tiempo.

Entenderemos *aprendizaje* como un proceso de optimización que puede ser visto como la búsqueda de una solución en un espacio de parámetros multidimensionales (pesos), que gradualmente optimiza una función objetivo predefinida. Existen diversas reglas de aprendizaje automático, como el aprendizaje supervisado, no supervisado, semisupervisado o por refuerzo. En la figura 4.1 podemos ver un esquema general de estos cuatro tipos de aprendizaje automático.

Nos centraremos en los dos primeros tipos de aprendizaje, puesto que ambos tipos fueron empleados por las metodologías predominantes en el estado del arte. Los trabajos de Nazar (2013) y Gérard et al. (2014) dan cuenta de este tipo de aprendizaje desde perspectivas distintas. En el caso de Nazar (2013), algoritmos de *clustering*, clasificación e inducción de significado. Y el caso de Gérard et al. (2014), máquinas de vectores de soporte y *topic modeling*.

En el *aprendizaje supervisado* cada patrón o señal de entrada recibida del ambiente es asociada con un patrón objetivo deseado. Generalmente los pesos son sintetizados gradualmente y, en cada paso del proceso de aprendizaje, son actualizados de forma que el error entre la red de salida y el objetivo deseado sean reducidos.

Por otra parte el *aprendizaje no supervisado* involucra técnicas como el *clustering*, o la detección de similitudes entre patrones no etiquetados en un conjunto de entrenamiento

determinado. La idea detrás de este tipo de aprendizaje es optimizar un criterio, o función de desempeño definida, en términos de la actividad de salida de las unidades de la red. Se espera que los pesos y las salidas de la red converjan en representaciones que capturen las regularidades estadísticas de los datos de entrada (Hassoun, 1995, p. 50).

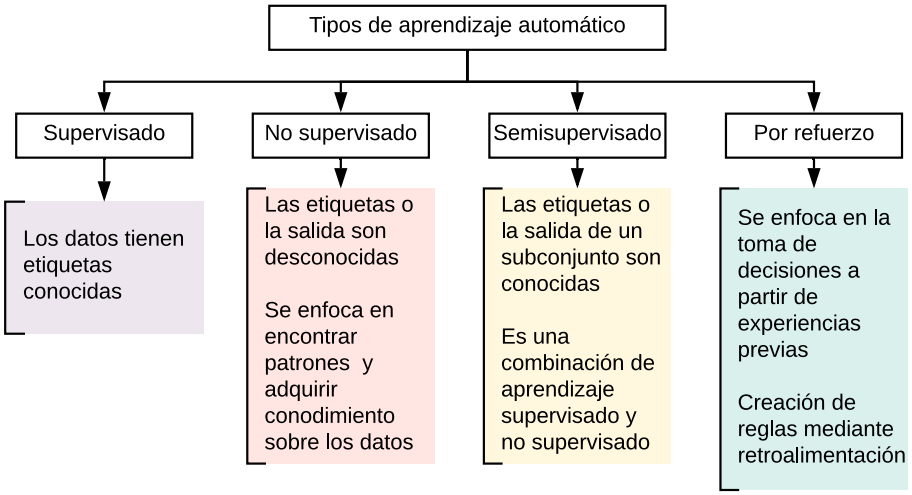


Figura 4.1 – Breve descripción de los diferentes tipos de aprendizaje automático.

El primer tipo de aprendizaje es el más interesante para este proyecto dado que se contempla entrenar modelos de clasificación de temas mediante ejemplos, en este caso corpus de lengua general y especializados. Los corpus de lengua general serán etiquetados gramaticalmente para obtener un modelo neuronal de cada lengua de trabajo, tendrán la función de desambiguar el significado de los candidatos a NS, mientras que los corpus especializados servirán para crear los modelos de clasificación de temas.

En cambio, el aprendizaje no supervisado funcionaría de la forma opuesta, es decir, mediante un conjunto de datos se espera encontrar patrones y sus relaciones. Este tipo de enfoque será aplicado en la etapa de extracción de palabras claves, puesto que el proceso de extracción de palabras claves depende del texto de entrada.

El funcionamiento y rol específico de cada parte del sistema serán explicados a detalle en el capítulo 5, que contiene la descripción de cada uno de los procesos y algoritmos utilizados por DENISE. En dicho capítulo también se describen los criterios de selección de corpus y la descripción de su procesamiento para llevar a cabo entrenamiento de los modelos antes mencionados.

4.2 La detección de neologismos semánticos por medio de estrategias de aprendizaje profundo

Como se ha mencionado en el capítulo anterior, los neologismos semánticos (NS) son un tipo de neologismo formado por una modificación del significado de la base léxica de una unidad, sin que la base altere su forma. Desde una perspectiva computacional entenderemos *neologismo semántico* como una cadena de caracteres que, dentro del cuerpo de un texto, se encuentra asociada a unidades de significación diferentes a las ya registradas en

los diccionarios de lengua general. Dichas unidades de significación extienden una temática y, por lo tanto, la discordancia entre temas puede ser un factor que ayude a detectar el proceso de cambio semántico.

El enfoque metodológico seleccionado consiste en afrontar la detección de neología semántica como un problema de clasificación (Allen, 1995; Hassoun, 1995), apoyándonos en métodos de aprendizaje automático de forma similar al trabajo realizado por Gérard et al. (2014). Se propone desarrollar un sistema capaz de clasificar las unidades de interés como neológicas, frente al conjunto conocido de significados (representado por un modelo neuronal de lengua general) y el conjunto conocido de temas (representado por un modelo de clasificación).

La desambiguación semántica es la habilidad de determinar, de forma computacional, el significado de una palabra que está siendo usada en un contexto determinado (Navigli, 2009). Mientras que nuestro enfoque toma elementos de dicho ámbito, tratamos la desambiguación de significado mediante un análisis de discordancia de temas: un NS pertenece a una temática distinta a la prototípica de una unidad candidata. Esta tarea se puede llevar a cabo mediante diferentes metodologías como árboles de decisión, clasificadores estadísticos, listas de decisiones, máquinas de soporte vectorial o redes neuronales, siento este último uno de los métodos implementados por DENISE.

En términos generales, una red neuronal se define como un sistema computacional construido por un número de elementos sencillos, altamente interconectados, que procesan información mediante la respuesta que tiene su estado dinámico ante estímulos externos (Caudill, 1987; Caudill y Butler, 1992; Vonk et al., 1995; Gurney, 1997; Fischer, 1998). En el caso de DENISE, el modelo de red neuronal Sense2Vec (Trask et al., 2015) se entrena usando corpus de lengua general etiquetados gramaticalmente. En este modelo, relaciones sintácticas que existen entre las palabras que conforman el vocabulario del corpus son los estímulos de los estados dinámicos de la red que producen como resultado *word emmbeddings* o representaciones vectoriales de palabras. La figura 4.2 muestra un ejemplo de las similitudes entre palabras que se pueden obtener mediante este tipo de modelos.

```
>>> model.most_similar(['Donald_Trump|PERSON'])
(u'Sarah_Palin|PERSON', 0.854670465),
(u'Mitt_Romney|PERSON', 0.8245523572),
(u'Barrack_Obama|PERSON', 0.808201313),
(u'Bill_Clinton|PERSON', 0.8045649529),
(u'Oprah|GPE', 0.8042222261),
(u'Paris_Hilton|ORG', 0.7962667942),
(u'Oprah_Winfrey|PERSON', 0.7941152453),
(u'Stephen_Colbert|PERSON', 0.7926792502),
(u'Herman_Cain|PERSON', 0.7869615555),
(u'Michael_Moore|PERSON', 0.7835546732)]
```

Figura 4.2 – Ejemplo de funcionamiento de Sense2Vec de la librería spaCy.

Este acercamiento toma como base el modelo Word2Vec propuesto por Mikolov et al. (2013c,a,b); Le y Mikolov (2014), con la adición del uso de etiquetas gramaticales para

desambiguar significados con mayor precisión. Como se puede ver en la figura 4.2, “Donald Trump” tiene bastante relación con “Sarah Palin”, “Mitt Romney”, “Barack Obama” y otras personalidades de la política estadounidenses. También nos muestra que, en la mayoría de los casos, son personas (*PERSON*) y finalmente indica el grado de similitud, o relación semántica, que existe entre “Donald Trump” y cada una de las unidades que componen el listado generado.

En este ejemplo, “Donald Trump”, corresponde correctamente a una persona y al mundo de la política. DENISE intenta detectar NS cuándo no existe esta concordancia, es decir, si en un contexto de entrada aparece una unidad cuya temática es distinta a su temática prototípica (determinada por los elementos de la lista), entonces dicha unidad es un candidato válido a NS. Esta disociación implica que una palabra existe previamente en el modelo de lengua general, pero pertenece a un campo del conocimiento diferente a los ya documentados dentro del modelo. Mientras este tipo de modelos es tolerante al ruido, esta particularidad permite que los modelos de lengua basados en arquitecturas neuronales sean más robustos en comparación con los modelos simbólicos. Podemos ver ejemplos de tolerancia al ruido en casos como: la generación de representaciones de unidades con errores ortográficos (*Barrack Obama* por *Barack Obama*) y en la asignación de etiquetas gramaticales incorrectas (*Oprah* con etiqueta GPE de región geográfica y *Paris Hilton* con etiqueta ORG de compañía).

4.3 Diagrama de flujo de procesos del sistema DENISE

DENISE es un sistema modular ya que cada una de las piezas que lo conforman puede ser reemplazada o mejorada sin afectar el resto de los componentes. En la figura 4.3 se muestran los componentes principales del sistema, así como su secuencia lógica dentro del programa.

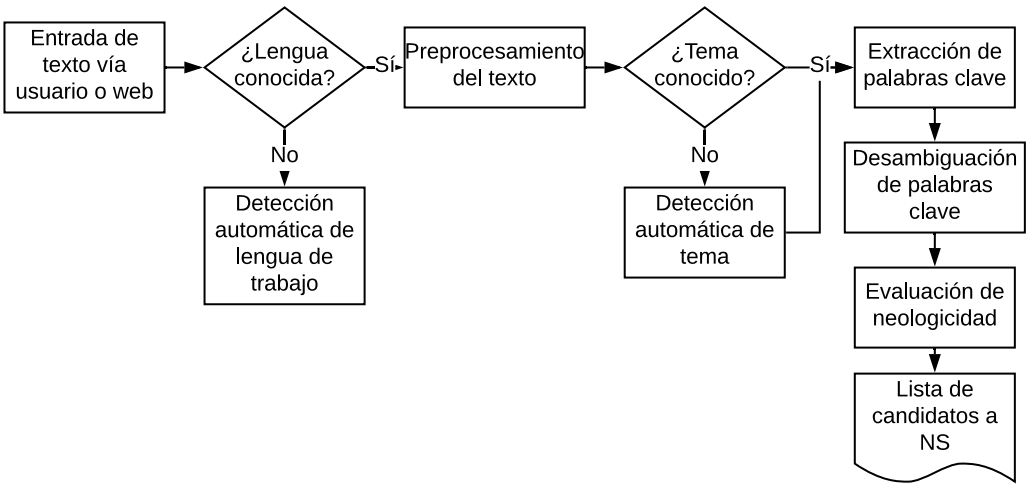


Figura 4.3 – Diagrama de flujo general del sistema DENISE.

El flujo de proceso inicia con la entrada del texto que será analizado, si el usuario puede seleccionar manualmente la lengua de trabajo o, en el caso de que no se conozca la lengua de trabajo del documento que será analizado, el sistema procede a detectar la lengua de trabajo automáticamente. Posteriormente se realizan tareas de preprocesamiento

de texto con la librería spaCy[4] de Python.

Una vez que el texto de entrada ha sido procesado, el siguiente paso es determinar su temática. En español y francés se pueden detectar los temas siguientes: informática, deportes y economía; mientras que en catalán se cuenta con los temas: derecho, economía, informática, lingüística, medio ambiente y medicina. El usuario puede seleccionar manualmente uno de los temas disponibles o dejar que el sistema detecte automáticamente la temática.

Después de haber detectado el tema principal, se extraen las palabras claves contenidas en el texto ingresado para generar una lista de precandidatos a NS. Con cada unidad del listado se realiza una consulta al modelo de lengua para obtener las palabras más similares, que servirán para delimitar el campo semántico de cada unidad consultada. El sistema procede a analizar la concordancia de temas entre cada campo semántico y el tema del texto ingresado, para presentar los posibles candidatos a NS. Una vez que ha finalizado el análisis, DENISE muestra los resultados de cada paso para que el usuario realice una selección final de candidatos.

En el capítulo 5 se explican las diferencias entre los diversos modelos de clasificación y modelos para la generación de *embeddings*. También se describen los procesos de entrenamiento que se siguieron para generar los modelos de clasificación y los modelos de representaciones vectoriales de palabras en cada lengua de trabajo de DENISE. Finalmente se presentarán las evaluaciones iniciales para llevar a cabo la selección de métodos y modelos definitivos.

[4] `https://spacy.io`

Capítulo 5

DESCRIPCIÓN DEL SISTEMA

El presente capítulo intenta presentar de forma detallada los elementos que conforman el sistema propuesto en esta tesis. Inicia con la descripción de un acercamiento previo a la detección de NS, en la que se exploran las ventajas y desventajas de un enfoque centrado en el uso de medidas de similitud. Posteriormente, en la sección 5.2, se describen los elementos que fueron requeridos para desarrollar nuestra metodología.

El contenido que abarca desde la sección 5.3 hasta la sección 5.7 contiene la descripción de los elementos principales y módulos opcionales que conforman nuestro sistema: un modelo de representación de documentos TF-IDF, un modelo de clasificación automática, una metodología para la extracción de palabras claves y modelos de aprendizaje profundo para generar representaciones distribuidas de palabras. En cada apartado explicamos el funcionamiento de cada elemento, los resultados de su evaluación y una justificación de su implementación.

Para evaluar los modelos clasificadores (así como otros casos tratados como problemas de clasificación) empleamos las métricas siguientes: exhaustividad, precisión, *f1-score*, soporte y exactitud. El objetivo de un clasificador es predecir correctamente la clase real que corresponde a un conjunto de datos. Como resultado de este proceso podemos obtener: verdaderos positivos (vp) cuando el resultado es correcto; verdaderos negativos (vn) cuando se clasifica correctamente la ausencia de un resultado; falsos positivos (fp) cuando se evalúa como correcta una clase asignada equivocada y falso negativos (fn) cuando se evalúa como incorrecta una clase asignada verdadera.

La precisión (ecuación 5.1) evalúa la fracción de documentos recuperados que han sido relevantes.

$$\text{precisión} = \frac{vp}{vp + fp} \tag{5.1}$$

La exhaustividad (ecuación 5.2) evalúa la fracción de documentos relevantes que han sido recuperados correctamente.

$$\text{exhaustividad} = \frac{vp}{vp + fn} \tag{5.2}$$

El *f1-score* (ver ecuación 5.3) puede ser definido como la media armónica de la precisión y la exhaustividad con un valor $\beta = 1$.

$$f_\beta = (1 + \beta^2) = \frac{\text{precisión} \times \text{exhaustividad}}{\beta^2 \text{precisión} + \text{exhaustividad}} \tag{5.3}$$

La exactitud evalúa la fracción de los casos que han sido clasificados correctamente. Cuando se trata de un clasificador binario empleamos la ecuación 5.4.

$$\text{exactitud} = \frac{vp + vn}{vp + vn + fp + fn} \tag{5.4}$$

Mientras que en el caso de clasificadores multiclase calculamos la exactitud mediante la ecuación 5.5, que puede ser expresada como la fracción de las predicciones correctas sobre la cantidad de $n_{muestras}$, donde y es el valor verdadero esperado e $\hat{y}$ el resultado del clasificador.

$$\text{exactitud}(y, \hat{y}) = \frac{1}{n_{muestras}} \sum_{i=0}^{n_{muestras}-1} 1(\hat{y}_i = y_i) \tag{5.5}$$

5.1 Un enfoque preliminar a la detección de la neología semántica mediante medidas de similitud

El primer acercamiento propuesto para la detección de NS consistió en un sistema que tomaba como entrada una lista de palabras candidatas a NS y utilizaba medidas de similitud para comparar —simultáneamente— sus contextos de aparición o concordancias, las acepciones de los diccionarios de referencia y un campo semántico que delimitaba un campo del conocimiento especializado. El resultado esperado de esta comparación era determinar si una palabra es candidata a NS en función de la similitud que puede tener su contexto de aparición.

La intención de esta comparación era determinar si un contexto (texto de entrada) es cercano o similar al conjunto del conocimiento de lengua general representado por los diccionarios de referencia o si, en cambio, es similar al campo semántico. Un índice de similitud mayor entre las acepciones del diccionario y el contexto, indicaría que el contexto en cuestión no aporta información novedosa. En cambio, si el contexto evaluado presenta mayor similitud con respecto al campo semántico, existe una probabilidad elevada de que dicho contexto tenga un significado neológico.

Se empleó un corpus especializado para realizar la extracción de términos y generar un listado de términos que pertenecen al campo de la informática. Este listado tiene una doble función: sirve como lista de referencia para seleccionar los términos de informática que se encuentran en la tabla de neologismos de OBNEO, así como para crear un campo semántico relacionado con esta misma temática.

El campo semántico generado debería aportar información de lengua general y especializada de forma balanceada, tal que nuestro algoritmo sea capaz de calcular la similitud entre el contexto que está siendo evaluado y las entradas de los diccionarios de referencia. A su vez, este listado de términos se incrementó usando los diccionarios terminológicos

de TERMCAT[1] que se encuentran disponibles para el público general.

Durante la primera etapa de procesamiento del sistema se valida que cada unidad a detectar se encuentre registrada en los diccionarios de referencia. Posteriormente, se eliminan los verbos definitorios de las definiciones que se obtuvieron y del contexto de la unidad a analizar. Los textos resultantes son interpretados como representaciones vectoriales para llevar a cabo la evaluación de similitud. Llevar a cabo esta tarea con contextos cortos es complicado, pero enfoques como el de Molina et al. (2010) apoyado en las ideas de Torres-Moreno (2011) permitieron trabajar con contextos cortos y obtener buenos resultados en resumen automático.

Después de realizar las evaluaciones de similitud, el sistema muestra al usuario los índices de similitud que tiene cada contexto frente a cada acepción registrada en los diccionarios, así como frente al campo semántico. Con esta información el usuario puede determinar si una unidad es un candidato válido a NS y en relación a qué acepción de cada uno de los diccionarios de referencia podemos interpretar un cambio de significado.

5.1.1 Recursos y descripción del sistema

La intención de este primer acercamiento ha sido que los elementos de trabajo del sistema fuesen flexibles, de forma que puedan ser reemplazados como piezas intercambiables. Esta es una de las características principales del diseño de esta aplicación. La primera implementación del sistema funciona en español, pero existe la posibilidad de añadir cualquier lengua de trabajo, mientras se cuente con todos los elementos requeridos por esta metodología. Los elementos usados por el sistema son los siguientes:

- Corpus generales para realizar consultas y obtener contextos para ser analizados.
- Corpus especializados para construir campos semánticos que delimiten las temáticas que pueden ser analizadas.
- Diccionarios de referencia para obtener definiciones y representar el vocabulario de la lengua general.
- Listas de términos y NS previamente detectados para realizar consultas y evaluaciones.

En la tabla 5.1 se pueden observar los corpus de trabajo que fueron usados, así como su tamaño en palabras en bruto. Los corpus generales sirven para obtener concordancias de los términos que son consultados. Se espera que estos contextos de aparición tengan información diferente a la contenida en las definiciones de los diccionarios de referencia, es decir, una unidad se puede considerar candidata a NS si su concordancia en los corpus de lengua general da cuenta de una realidad diferente a la documentada en los diccionarios.

Por otra parte, los corpus especializados sirvieron para realizar la extracción de términos de informática y así crear un campo semántico que englobe los términos más representativos e informativos de este campo del conocimiento. Este proceso se llevó a cabo analizando la frecuencia de cada término en el corpus ponderándola con el número de resultados en Google. La extensión del campo semántico también se limitó a 150 palabras,

[1] `http://www.termcat.cat/es/Diccionaris_En_Linia/`

Corpus	Lengua	Tipo	Tamaño
PC Word	ES	Especializado	36,397,039
Wikipedia Informática	ES	Especializado	79,748,494
Wikipedia Informática	FR	Especializado	56,408,978
Jornada	ES	General	723,528,525
El Financiero	ES	General	230,111,744
Wikipédia	FR	General	3,713,159,148
L'Observateur	FR	General	1,300,687,386
L'Est Républicain	FR	General	790,616,360
Le Monde	FR	General	347,531,455
VSD	FR	General	65,408,978

Tabla 5.1 – Extensión en palabras de los corpus de trabajo.

puesto que se encontró que un número menor de palabras no era suficiente para calcular la similitud y un número mayor a 150 tendía a favorecer la similitud con el campo semántico.

Los diccionarios de referencia que usa el sistema son: el *Diccionario del español de México* (DEM[2]) del Colegio de México, el *Diccionario de la lengua española* de la Real Academia Española (DLE[3]) y el *Wiktionary* (WIKI[4]) en español. Todos estos diccionarios han sido procesados en forma de una tabla CSV para facilitar el procesamiento informático. Sus acepciones sirven para determinar si los términos consultados están documentados (de forma similar al criterio lexicográfico) y, en caso de estarlo, calcular el grado de similitud de cada acepción con la concordancia de corpus general.

Para realizar los cálculos mencionados anteriormente, este enfoque combina el uso de medidas de similitud y representación de textos en espacios vectoriales. El uso de espacios permite una representación adecuada de los textos en forma matricial. Esta matriz consiste en una transformación de los textos en vectores con números reales, donde cada palabra representa un elemento de un vector y cada elemento evaluado (campo semántico, acepción y contexto de entrada) representa un vector de la matriz de similitud.

Esta representación vectorial se emplea para calcular el índice de neologicidad de un candidato de la siguiente manera: cuando dicho índice tiende hacia un valor cercano a 1, una palabra tiene mayor probabilidad a ser un candidato a NS; en cambio, cuando este índice tiende a 0, el término analizado no se considera un candidato viable y es descartado.

La medida coseno (ecuación 5.6) es una medida que permite calcular el grado de similitud que existe entre dos vectores en función de los atributos que los conforman (Baeza-Yates y Ribeiro-Neto, 1999; Singhal, 2001; Tan et al., 2005). En este caso la entrada es una cadena de texto que se transforma en un espacio vectorial, de forma que el coseno euclidiano pueda ser usado para determinar la similitud entre textos. Esta función calcula la similitud entre dos vectores de entrada $\mathbf{A}$ y $\mathbf{B}$ mediante el producto escalar y la magnitud de sus componentes A_i y B_i.

Un resultado de similitud coseno que tiende a 1 un indica que la similitud entre vectores es exacta, por ejemplo, la similitud de un vector con sí mismo daría como resultado 1.

[2] `https://dem.colmex.mx`
[3] `https://dle.rae.es`
[4] `https://es.wiktionary.org/`

Por otra parte, un resultado que tiende a (o es menor que) 0 indica que los espacios vectoriales son opuestos y por lo tanto no existe similitud entre los textos que están siendo analizados.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}|| \, ||\mathbf{B}||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{5.6}$$

El sistema propuesto (ver figura 5.1) toma como entrada una palabra o lista de palabras. El usuario selecciona la lengua de trabajo y uno de los campos semánticos disponibles, en este caso, informática. A continuación, se lleva a cabo un primer proceso de filtrado, si la palabra ingresada no existe en los diccionarios de referencia se interrumpe el proceso y se presenta como candidato a neologismo de forma. En el caso contrario, se extraen las acepciones existentes en los diccionarios de referencia.

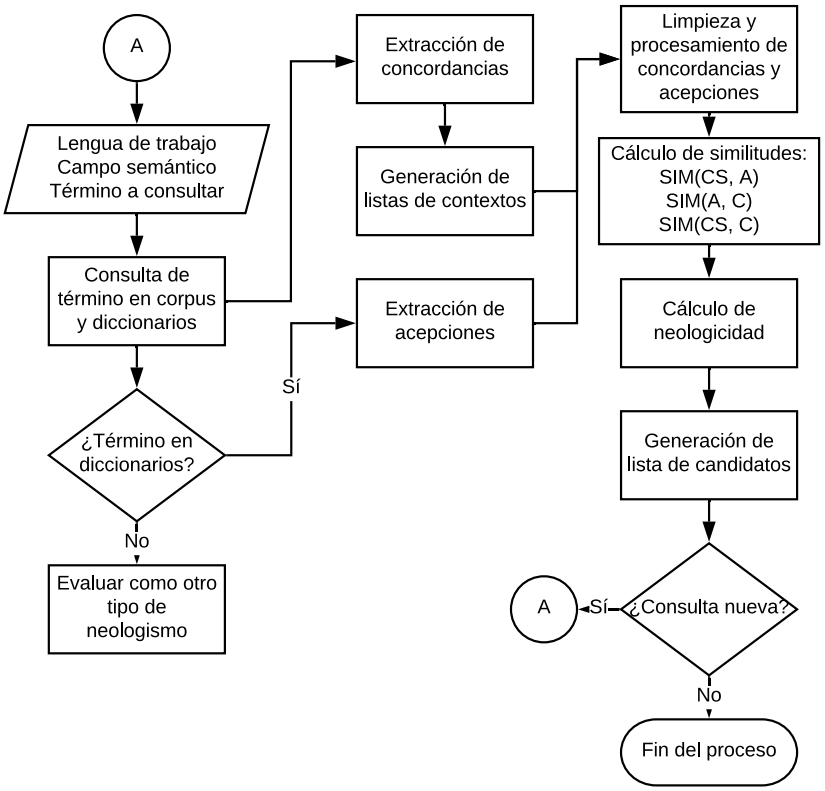


Figura 5.1 – Diagrama de flujo de la primera propuesta del sistema DENISE.

Paralelamente a los procesos mencionados, el sistema consulta los corpus de lengua general para extraer las concordancias de cada palabra ingresada. Después realiza las siguientes tareas de preprocesamiento: limpieza y normalización del texto, tokenización, eliminación de stopwords, segmentación de textos con el módulo segmentador de Cortex (Torres-Moreno et al., 2001, 2002) y stemming. Una vez que se han procesado todos los textos de entrada, éstos son transformados en representaciones vectoriales y se procede a calcular las siguientes similitudes:

- Similitud entre la acepción del diccionario y el campo semántico: $sim(CS, A)$.
- Similitud entre la acepción del diccionario y el contexto: $sim(A, C)$.

- Similitud entre el contexto ingresado y el campo semántico: $sim(CS, C)$.

Bajo los supuestos anteriores *sim* corresponde al cálculo de la matriz de similitud mediante el coseno euclidiano, *CS* representa el campo semántico de informática, *A* una acepción de diccionario y *C* la concordancia obtenida de los corpus de lengua general de la palabra que ha sido consultada por el usuario.

El proceso de análisis final, el análisis de neologicidad (NEO_s), toma en cuenta los resultados de los cálculos de la matriz de similitud para determinar si alguno de los tokens de las concordancias analizadas puede ser un candidato válido a NS. Un candidato ideal a NS es aquel elemento consultado que cumple con las siguientes condiciones: cuenta con un alto índice similitud con el campo semántico seleccionado y un bajo índice de similitud con las acepciones de los diccionarios y, simultáneamente, el campo semántico tiene un bajo índice similitud con las acepciones de los diccionarios.

Estas condiciones indican que la concordancia del término consultado describe un significado novedoso, distinto a los ya documentados en las acepciones que se encuentran en los diccionarios. El caso opuesto indicaría que el elemento consultado ya cuenta con una acepción en diccionarios que da cuenta de la realidad que representa, la concordancia es similar a esta acepción y, por lo tanto, se considera un elemento no neológico.

5.1.2 Evaluación

Como metodología de evaluación de esta primera propuesta empleamos la validación de los contextos de la base de datos de OBNEO (contextos de prueba) que corresponden al campo de la informática. Estos contextos cuentan con neologismos previamente detectados que el sistema debe evaluar como candidatos válidos. Simultáneamente se lleva a cabo un segundo proceso de validación con contextos nuevos obtenidos de los corpus generales (contextos de análisis). Esta segunda validación sirve para medir la eficacia con contextos que el sistema no ha analizado anteriormente.

Se ingresaron 35 NS seleccionados manualmente de la base de datos de OBNEO, cuya característica principal es que todos cuentan con acepciones en nuestros tres diccionarios de referencia[5]. El objetivo de este experimento era comprobar que son evaluados como neológicos frente a las acepciones de lengua general y simultáneamente son evaluados como no neológicos frente a las acepciones informáticas. Estos 35 NS cumplen con los siguientes criterios de selección:

- Han sido evaluados manualmente en OBNEO.
- Cuentan con definiciones en los tres diccionarios de referencia.
- Pertenecen al campo de la informática.
- Pueden obtenerse nuevos contextos en nuestros corpus para la segunda evaluación.

[5]En consecuencia, se consideran no neológicos en la actualidad, no obstante, dado que la finalidad de este estudio es corroborar la eficacia del sistema, nos interesa tener elementos estables en la lengua para comprobar que los cálculos de similitud se realizan de forma correcta y que el sistema es capaz de determinar si un contexto de entrada es novedoso y distinto a los significados de las acepciones recogidas en los diccionarios.

En la figura 5.2 se muestra el listado de los 35 términos y la similitud acumulada con el campo semántico $sim(CS, C)$, esto nos ayuda a visualizar rápidamente los posibles términos que tienen mayor similitud con el campo semántico y en la figura 5.3 podemos ver el listado de los 35 términos y la similitud acumulada que existe entre el campo semántico y las acepciones de los diccionarios $sim(CS, A)$.

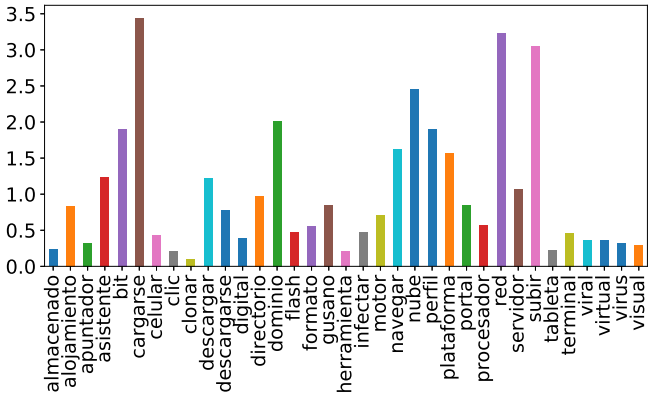


Figura 5.2 – Similitud acumulada por elemento entre concordancias y campo semántico.

En la figura 5.3 se observa el listado de los 35 NS y la similitud acumulada de los contextos obtenidos para cada NS y las acepciones de los diccionarios. Estos datos muestran de forma resumida la relación que existe entre los elementos que son analizados, sin embargo, son poco informativos ya que no podemos observar frente a qué acepciones un candidato es neológico, o si todos los diccionarios cuentan con la misma cantidad de acepciones.

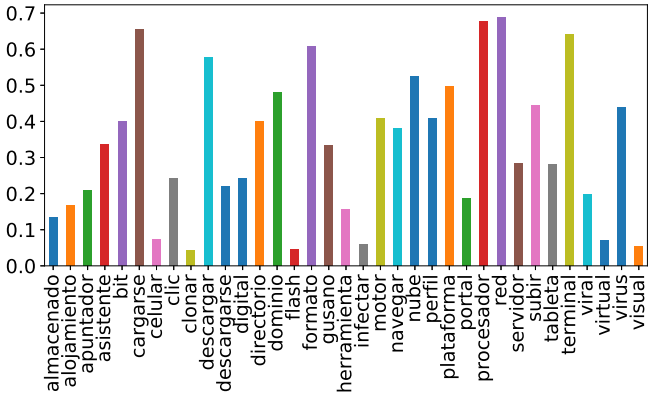


Figura 5.3 – Similitud acumulada por elemento entre acepciones de los diccionarios de referencia y campo semántico.

Los 35 términos consultados generaron 563 casos para ser analizados, de antemano se sabe que estos términos han sido clasificados manualmente como NS. Como resultados de este proceso podemos esperar *Falso* (0) en el caso de los significados informáticas y *Verdadero* (1) frente a los significados no neológicos. En la figura 5.4 se muestran los resultados acumulados por término–concordancia y acepciones de los diccionarios.

Las métricas empleadas para el cálculo de la neologicidad toman en cuenta los resultados del cálculo de la similitud coseno y presentan un resultado de *Verdadero* (1) si

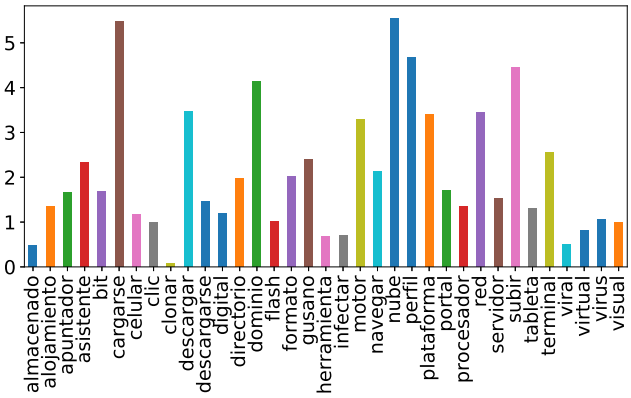


Figura 5.4 – Similitud acumulada por elemento entre concordancias y acepciones de los diccionarios de referencia.

un contexto es considerado neológico o *Falso* (0) si un contexto es evaluado como no neológico. Se introdujeron tres metodologías para obtener candidatos a NS, la primera la ecuación 5.7, donde $\mathrm{sim}(x, y)$ son las distintas combinaciones del cálculo de la similitud por coseno euclidiano; CS representa el campo semántico o lista de términos informáticos, A la acepción obtenida del diccionario de referencia y C la concordancia donde aparece un posible candidato. Esta ecuación tiende a dar preferencia a las similitudes con el campo semántico.

$$\mathrm{ns}_{JM} = sim(CS, C) + \frac{(1 - sim(A, C)) + (1 - sim(CS, A))}{3} \tag{5.7}$$

Por otra parte, la ecuación 5.8, tiende a dar preferencia a las diferencias entre similitudes de la lista de términos informáticos CS, la concordancia analizada C y la acepción del diccionario A.

$$\mathrm{ns}_{AT} = (1 + sim(CS, C)) - (1 + (sim(A, C) + sim(CS, A))) \tag{5.8}$$

El tercer caso es una función que retorna *Verdadero* cuando la similitud entre CS y C es mayor que la similitud entre A y C y, simultáneamente, la similitud entre CS y C es mayor que la similitud entre CS y A. En el caso de que no se cumplan las condiciones antes mencionadas, la función retorna *Falso*.

$$\mathrm{candidato} = \begin{cases} Verdadero & \text{si } (sim(CS, C) > sim(A, C)) \ \& \\ & (sim(CS, C) > sim(CS, A)) \\ Falso & \text{en otro caso} \end{cases} \tag{5.9}$$

5.1.3 Resultados

Para validar los resultados obtenidos con las fórmulas para el cálculo de neologicidad se emplearon el coeficiente kappa de Cohen (Cohen, 1960; Fleiss et al., 1969; Banerjee et al.,

1999) y la correlación de Pearson (Pearson, 1895), además de las medidas de precisión, exhaustividad, *f-Score* y soporte para obtener una visión más clara de la efectividad del sistema. Para calcular estos últimos índices se compararon los resultados con los casos que fueron evaluados manualmente (los contextos obtenidos de la base de datos de OBNEO). En la figura 5.5 se puede observar un fragmento de los resultados generados por el sistema, con ver los contextos analizados y los valores obtenidos.

	term_id	term	context	dic	dic_id	definition	man_cond	CS_Tn	CS_DIC	DIC_Tn	ns_AT	ns_JM	shape	cond
0	0	almacenado	esta transmisión de conocimiento dentro de la ...	dle	0	poner o guardar en almacén	1	0.128628	0.000000	0.053410	0.075218	0.110825	(3, 128)	1
1	0	almacenado	esta transmisión de conocimiento dentro de la ...	dle	0	registrar información en la memoria de un orde...	0	0.110489	0.099474	0.245637	-0.234622	0.061768	(3, 127)	0
2	0	almacenado	esta transmisión de conocimiento dentro de la ...	dle	0	reunir o guardar muchas cosas	1	0.127808	0.000000	0.000000	0.127808	0.127808	(3, 129)	1
3	0	almacenado	esta transmisión de conocimiento dentro de la ...	dem	1	que está guardado en un lugar o espacio, gener...	1	0.130315	0.000000	0.084513	0.045802	0.102144	(3, 137)	1
4	0	almacenado	esta transmisión de conocimiento dentro de la ...	wiki	2	participio de almacenar.	1	0.108840	0.037733	0.321765	-0.250658	0.014162	(3, 126)	0

Figura 5.5 – Fragmento de resultados de la evaluación.

En la tabla 5.2 señalamos en negritas el resultado de correlación de Pearson que existe entre los valores que fueron evaluados manualmente (*man_cond*) y la métrica *ns_AT*, esta es la correlación más alta entre las fórmulas implementadas. A pesar de que se obtuvo un valor inferior a 0.5, cifra que se suele usar como referencia para determinar que existe correlación, sigue siendo el más alto en comparación con los demás valores obtenidos.

	ns_JM	ns_AT	cond	man_cond	CS_Tn
ns_JM	1.0000	0.7681	0.6555	-0.0504	0.6197
ns_AT	0.7681	1.0000	0.7163	0.2269	0.2127
cond	0.6555	0.7163	1.0000	0.1548	0.3658
man_cond	-0.0504	**0.2269**	0.1548	1.0000	0.1439
CS_Tn	0.6197	0.2127	0.3658	0.1439	1.0000

Tabla 5.2 – Correlación de Pearson entre métricas y contextos evaluados manualmente.

Por otra parte, los resultados obtenidos con el coeficiente kappa de Cohen denotan la concordancia que existe entre evaluadores, en este caso, las métricas *cond* y *man_cond*. Entre estas dos métricas se obtuvo un resultado de 0.0512, lo que indica que existe una leve correlación entre resultados. En términos de kappa un resultado inferior a 0 indica que no existe concordancia entre evaluadores, mientras que un valor de kappa entre 0 y 0.2 puede indicar la existencia de una leve correlación entre los sistemas de evaluación comparados.

Dado que nuestra métrica *cond* depende de condiciones lógicas (tener mayor similitud con el campo y menor similitud con las acepciones del diccionario), da como resultado un valor booleano *verdadero* (1) o *falso* (0). Se usaron estos resultados para realizar los cálculos de precisión, exhaustividad, *f1-score* y soporte, los resultados se pueden encontrar en la tabla 5.3.

La precisión es la habilidad que tiene el modelo de clasificación de clasificar de forma correcta los elementos ingresados, no obstante, esta habilidad no necesariamente se

corresponde con un alto índice de resultados correctos. En el caso de la precisión de la métrica *cond*, nuestro sistema tiene un 0.99 (entendido como un 99 %) de precisión, en cambio, el resto de los índices muestra que este no es un resultado que pueda ser empleado como referencia.

	man_cond	cond
Precisión	0.1007	0.9931
Exhaustividad	0.9767	0.2788
f1-Score	0.1826	0.4354
Soporte	43	520

Tabla 5.3 – Precisión, exhaustividad, *f1-score* y soporte entre valores etiquetados manualmente y valores etiquetados automáticamente por el sistema.

La exhaustividad indica la cantidad de verdaderos positivos evaluados correctamente, mientras esta metodología que obtuvo una precisión de 0.99, tiene un 0.27 de exhaustividad, esto equivale a un 30 % de casos de verdaderos positivos evaluados de forma correcta. El *f1-score* representa la media geométrica ponderada de la precesión y exhaustividad, este índice toma en cuenta todos los resultados, tanto los falsos positivos como los verdaderos positivos. En nuestro caso obtuvimos un *f1-score* de 0.43.

Si se analizan los casos de verdaderos positivos (ver tabla 5.4), se puede comprobar que los índices de neologicidad de la fórmula *ns_AT* tienen las correlaciones esperadas. El resultado tiende hacia 1 cuando se evalúa frente a acepciones de lengua general y tiende hacia -1 cuando se evalúa frente a una definición informática. Al existir un índice que tiende a -1 el candidato es descartado.

	ns_JM	ns_AT	cond	man_cond	CS_Tn
ns_JM	1.0000	0.6223	0.4666	0.4666	0.8795
ns_AT	0.6223	1.0000	0.8852	0.8852	0.4655
cond	0.4666	**0.8852**	1.0000	1.0000	0.4346
man_cond	0.4666	**0.8852**	1.0000	1.0000	0.4346
CS_Tn	0.8795	0.4655	0.4346	0.4346	1.0000

Tabla 5.4 – Correlación de Pearson entre métricas para casos evaluados correctamente.

La correlación de Pearson para casos de verdaderos positivos muestra que el cálculo de *ns_AT* incrementa frente a los índices evaluados tanto automática como manualmente. Por lo tanto, se puede concluir que esta fórmula ha sido la más eficaz para realizar el análisis de neologicidad, en comparación con el resto. Los resultados iguales a 1 representan la correlación que tiene una métrica consigo misma.

5.1.4 Limitaciones y mejoras

Como se puede observar en la figura 5.6, el diccionario de la lengua española cuenta con más acepciones por entrada que el resto de los diccionarios empleados. Esta característica desequilibra las matrices de similitud ya que muchas de sus definiciones son tautológicas,

sinonímicas o referencias a otras entradas del diccionario. Estos tipos de definiciones son poco informativas y provocan un incremento de resultados de falsos negativos, ya que las matrices de similitud no cuentan con los suficientes elementos en común entre vectores para realizar el cálculo de la similitud.

Una posible solución a este problema podría ser ajustar la muestra de acepciones de cada diccionario, en vista de que el DLE cuenta un mayor número de acepciones en comparación con los demás diccionarios de referencia utilizados y, como se ha mencionado anteriormente, no todas sus acepciones son lo suficiente informativas. Este ajuste implicaría determinar un número de acepciones por entrada y determinar una longitud mínima en palabras por cada acepción. Ambas soluciones conllevan otros problemas, por ejemplo: eliminar acepciones que solamente están registradas en un diccionario de referencia o eliminar acepciones de los tres diccionarios.

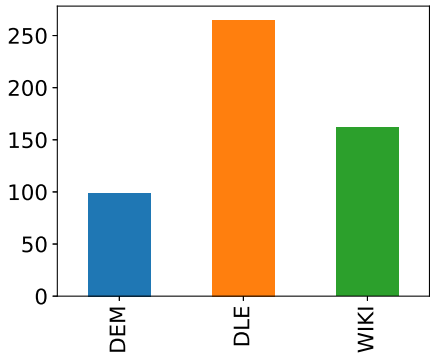


Figura 5.6 – Distribución de acepciones por diccionario de referencia.

Con respecto a las métricas para el cálculo de la neologicidad, las fórmulas propuestas dependen del cálculo de similitud de coseno, pero solamente en un nivel estrictamente formal, es decir, solamente se está evaluando la similitud palabra a palabra entre las acepciones de los diccionarios de referencia, el campo semántico y el contexto de entrada. Como mejora a este cálculo se podría considerar agregar pesos adicionales, por ejemplo: la probabilidad de distribución de la matriz de similitud, la fuente de las acepciones y contextos de entrada, las longitudes de cada definición, entre otros elementos formales que puedan ser más informativos.

En resumen, los resultados obtenidos con esta primera propuesta tienden a ser positivos, aunque solo moderadamente. Los cambios mencionados podrían mejorar los resultados, la depuración de las acepciones, hipotéticamente, debería incrementar las cifras observadas en exhaustividad y *f1-score*. No obstante, estas adecuaciones serían soluciones parciales, ya que la interacción del usuario se limitaría a validar que una palabra observada en un texto puede ser un candidato a NS.

Por lo tanto, en función de los resultados observados, se ha optado por un cambio de enfoque conservando las ideas fundamentales que se exponen en esta sección, la similitud y los espacios vectoriales, pero aplicadas con metodologías distintas: un sistema de detección automática de temas que determine, mediante similitud de representaciones de palabras en espacios vectoriales y un modelo de regresión logística, la temática de un texto de entrada.

También se añaden dos elementos nuevos: extracción automática de palabras claves

con filtros de etiquetas gramaticales mediante aprendizaje automático no supervisado (Mihalcea y Tarau, 2004; Mihalcea, 2004; Li y Wang, 2014; Barrios et al., 2016; Pay et al., 2018) y un método de desambiguación de significado por *word embedding* con etiquetas gramaticales (Trask et al., 2015) para extraer candidatos a NS desde un texto de entrada, con la intención de detectar automáticamente la existencia de posibles palabras que sean candidatas a NS en el texto de entrada.

El modelo de desambiguación se basa en el algoritmo de Word2Vec (Mikolov et al., 2013c) porque implementa el mismo procesamiento de entrenamiento neuronal, además de expandir el trabajo de Huang et al. (2012), dado que emplea un etiquetado gramatical supervisado. Este modelo simplifica los pasos a seguir, gracias a la implementación de estrategias más robustas.

Bajo esta nueva propuesta, los modelos neuronales de lengua cubren el rol de los diccionarios como método de desambiguación, ya que abren la posibilidad a encontrar relaciones semánticas en la totalidad de un corpus de lengua general y no solamente en lo contenido en las definiciones de los diccionarios. Por lo tanto, se espera poder diferenciar entre los significados de cada elemento existente en el vocabulario del modelo de lengua general y los nuevos significados pertenecientes a un ámbito especializado.

5.2 Un sistema de detección de neologismos semánticos mediante estrategias de aprendizaje profundo: DENISE

En los siguientes apartados se describirán todos los recursos que fueron necesarios para el desarrollo del sistema DENISE. Este sistema utiliza metodologías de aprendizaje automático supervisado y, por lo tanto, requiere datos de entrada para llevar a cabo las tareas de entrenamiento de los algoritmos y modelos.

Nuestros datos de entrenamiento se componen, principalmente, de corpus de lengua general para el entrenamiento de los modelos neuronales, mientras que los corpus especializados se usan para la clasificación de temas. Cabe señalar que, en el caso del catalán, los corpus de lengua general también fueron empleados para el desarrollo de un modelo de lengua similar a los de español y francés.

Para llevar a cabo las tareas de evaluación de los diferentes algoritmos generamos listas de NS y sus contextos de aparición, a partir de las bases de datos del OBNEO. Estas bases de datos tienen registro de todos los NS que se han detectado manualmente, en catalán y español, desde 1989 hasta 2015.

El proyecto depende de las siguientes librerías de Python: spaCy y NLTK[6] para realizar el preprocesamiento y etiquetado de textos; y scikit-learn[7] y gensim[8] para las tareas de aprendizaje automático y entrenamiento de modelos neuronales y de clasificación.

[6] `https://www.nltk.org`
[7] `https://scikit-learn.org/stable/`
[8] `https://radimrehurek.com/gensim/`

5.2.1 Corpus de trabajo

Los corpus de trabajo se dividen, pues, en dos categorías: corpus de lengua general y corpus especializados. Todos los corpus, tanto generales como especializados, se encuentran almacenados como archivos de texto plano con un enunciado por línea con codificación UTF-8 para considerar tildes y otras grafías propias de las lenguas de trabajo. Los corpus generales tienen versiones etiquetadas y no etiquetadas en un solo fichero mientras que los corpus especializados se dividen en un directorio para cada tema y un fichero para cada documento.

Los corpus de lengua general tienen dos roles principales: entrenamiento de modelos de aprendizaje automático y generación de modelos de lengua. En el caso del español y francés no hubo necesidad de crear modelos de lengua ya que se contaba con modelos previamente entrenados, en cambio, para el catalán hubo la necesidad de crear un modelo similar a los ya existentes en español y francés.

Como se ha mencionado, una de las funciones de los corpus de lengua general es realizar el entrenamiento de los modelos neuronales que serán empleados durante la fase de desambiguación. Puesto que las representaciones que generan estos algoritmos varían en función de los datos de entrada, usar Wikipedia nos permite crear representaciones equivalentes en cada una de las lenguas de trabajo.

Además de los *dumps* de Wikipedia[9], también se empleó el corpus anotado AnCora-CA en catalán (Taule et al., 2004) para crear un modelo de lengua catalana compatible con spaCy. Esta decisión se tomó puesto que el modelo, previamente existente, de lengua española fue generado y entrenado empleando el corpus AnCora-ES en español. La tabla 5.5 muestra el tamaño en palabras de cada uno de los corpus de lengua general.

Corpus	Lengua	Extensión en palabras
AnCora-CA	Catalán	500,500
Wikipedia CA	Catalán	118,327,537
Wikipedia ES	Español	397,166,701
Wikipedia FR	Francés	468,980,370

Tabla 5.5 – Tamaño en palabras de cada corpus de lengua general.

En el caso del catalán, empleamos el corpus AnCora-CA y spaCy para crear un modelo de lengua catalana[10] ya que spaCy cuenta con soporte integrado para español y francés, pero no para catalán. El modelo de lengua española de esta librería fue entrenado empleando el corpus AnCora-ES en español, un corpus de características similares a AnCora-CA, razón por la que se decidió emplear este mismo corpus y repetir el proceso de entrenamiento descrito en la documentación[11] de spaCy.

El modelo usa como entrada texto plano y como salida genera una matriz con los resultados del procesamiento. Otra de las razones para entrenar un modelo propio ha sido para asegurarnos que todos los corpus fueran procesados bajo las mismas condiciones y

[9] Utilizamos los *dumps* de Wikipedia preprocesados del proyecto Polyglot (Al-Rfou et al., 2013). Los *dumps* se encuentran disponibles en `https://sites.google.com/site/rmyeid/projects/polyglot`

[10] `https://github.com/bazzmx/spacy_catalan_model`

[11] `https://spacy.io/usage/training`

las mismas herramientas. En la tabla 5.6 podemos ver un ejemplo de la matriz resultante en forma de tabla.

Forma	Lema	Dependencia	POS	Vector
El	El	det	DET	12204527652707022206
Suprem	Suprem	nsubj	PROPN	16072095006890171862
decidirà	decidir	ROOT	VERB	13110060611322374290
el	ell	det	DET	4370460163704169311
5	5	obl	NUM	8148669997605808657
de	de	case	ADP	4370460163704169311
novembre	novembre	compound	NOUN	13110060611322374290
qui	qui	obj	PRON	4088098365541558500
paga	pagar	ccomp	VERB	13110060611322374290
finalment	finalment	advmod	ADV	13110060611322374290
l'impost	l'impost	obj	NOUN	6849787184664117039
sobre	sobrar	case	ADP	13110060611322374290
les	ell	det	DET	4088098365541558500
hipoteques	hipotecar	obl	NOUN	13110060611322374290

Tabla 5.6 – Ejemplo de elementos que forman la matriz resultante del procesamiento.

Por otra parte, la función de los corpus especializados consiste en crear y entrenar modelos de clasificación de documentos que serán usados como medio para la detección de temas. Dado que el alcance del sistema DENISE se limita a las unidades neológicas que provienen del campo de la informática, la clasificación se tratará como un problema binario, es decir, solamente se realiza la clasificación perteneciente a informática o no perteneciente a informática. Los temas que conforman los corpus especializados son los siguientes:

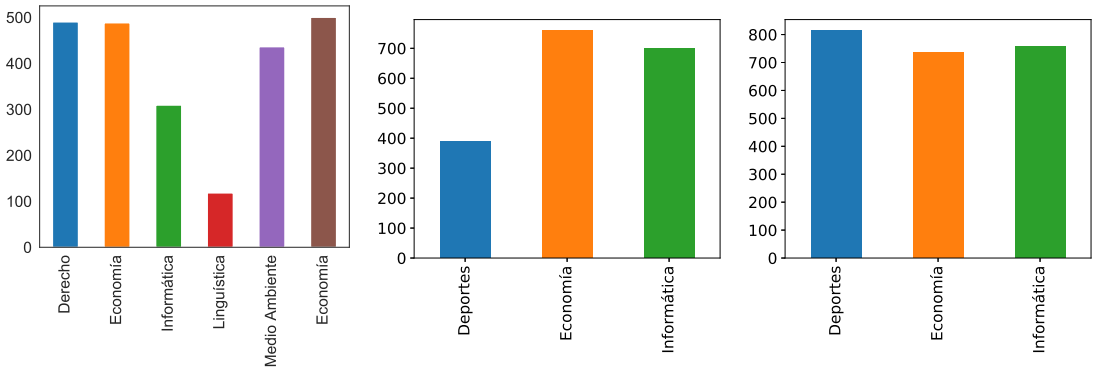
- Catalán: Informática, medicina, medio ambiente, economía, lingüística y derecho.
- Español: Informática, deportes y economía.
- Francés: Informática, deportes y economía.

En catalán se empleó el corpus técnico del IULA (en adelante CT-IULA) (Morel Santasusagna et al., 1998; Bach y Cabré, 2004; Cabré et al., 2006; Varga et al., 2007; Vivaldi, 2009), mientras que en español y francés se usaron *web scrappers* para obtener textos de prensa especializada con características similares a los textos del CT-IULA. En la tabla 5.7 se puede observar el tamaño en palabras de cada corpus especializado, así como las fuentes que fueron consultadas para compilarlos.

En español se extrajeron textos de la revista *PC World*, *Marca* y *El Financiero*, mientras que en francés se usaron las secciones correspondientes a tecnología, deportes y economía y finanzas del periódico *Le Monde*. En la figura 5.7 podemos ver la cantidad de documentos por temática en cada una de las lenguas de trabajo de DENISE.

Nombre	Lengua	Fuente	Extensión en palabras
PC World	ES	Revista	308,930
Marca	ES	Revista	275,872
El Financiero	ES	Prensa	280,404
CT-IULA	CA	Revista	7,569,596
Le Monde Sports	FR	Prensa	419,065
Le Monde Economie	FR	Prensa	362,791
Le Monde Informatique	FR	Prensa	325,347

Tabla 5.7 – Tamaño en palabras por corpus especializado.



(a) Catalán (b) Español (c) Francés

Figura 5.7 – Número de documentos por tema y lengua de trabajo.

5.2.2 Base de datos de neologismos del OBNEO

Como parte fundamental del proceso de evaluación del sistema, se emplearon listados de neologismos semánticos (acompañados de sus contextos de aparición) previamente etiquetados manualmente. Contar con datos validados manualmente nos permite tener un marco de referencia para realizar experimentos y evaluaciones de los diferentes procesos del sistema, principalmente clasificación de tema, extracción de palabras clave (KW) y selección de candidatos a NS.

A partir de la base de datos del OBNEO generamos dos tablas de NS, una en catalán y otra en español. Dichas tablas de neologismos abarcan el periodo comprendido desde 1989 hasta 2015 y ambas con las mismas características: incluyen datos como la fuente de la concordancia con la forma neológica y la categoría gramatical de cada NS etiquetada manualmente. En la figura 5.8 podemos observar la distribución de los NS por categoría gramatical en español y catalán. Por otra parte, en la figura 5.9 se muestra la distribución de NS por cada categoría gramatical concatenada.

Contar con las categorías gramaticales a las que pertenecen cada NS, nos permitirá descartar las categorías menos productivas (conjunciones, interjecciones, adverbios, etc.) durante el proceso de extracción de palabras claves y, de esta forma, limitar la extracción de palabras claves a aquellas que pertenecen a las categorías que concentran la mayoría de los NS registrados: sustantivos, verbos y adjetivos. Este mismo supuesto de trabajo se aplicará en las tres lenguas de trabajo.

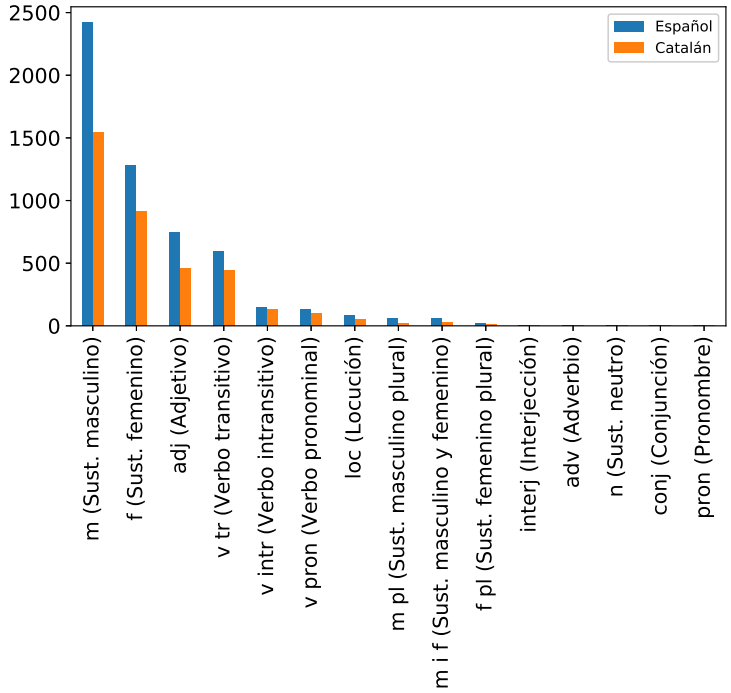


Figura 5.8 – NS por categoría gramatical en catalán y español.

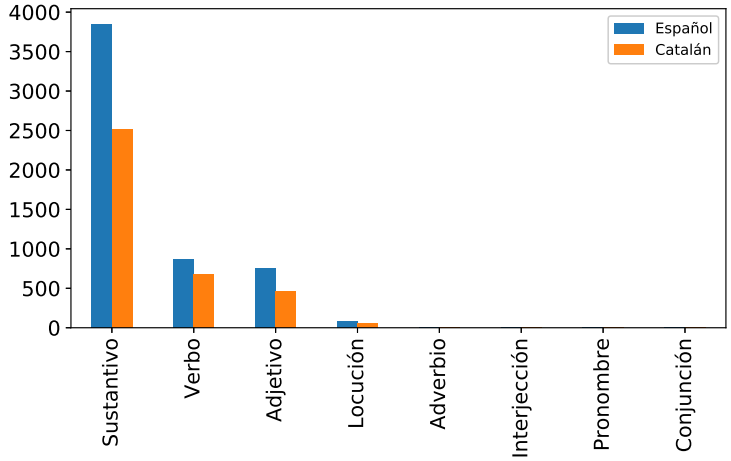


Figura 5.9 – NS por categoría gramatical concatenada en catalán y español.

5.3 Selección de lengua de trabajo

Durante la etapa de entrada de datos el usuario tiene la opción de introducir manualmente la lengua de trabajo de los textos de entrada o dejar que el sistema detecte la lengua de trabajo automáticamente. La detección se realiza mediante la implementación de Langdetect (Shuyo, 2010) para Python. Langdetect calcula la probabilidad de que un texto esté escrito en una lengua determinada mediante los atributos ortográficos y un clasificador bayesiano ingenuo.

Este algoritmo trata la detección de tema como un problema de clasificación de documentos, es decir, los documentos que conforman cada corpus de entrenamiento se cla-

sifican como cada lengua (C_k) que puede ser detectada. Con los documentos de entrenamiento crea perfiles de lengua con las probabilidades ortográficas de cada uno mediante la generación de n-gramas a nivel de palabra.

Tal como se muestra en la ecuación 5.10, Langdetect calcula la probabilidad que tiene cada uno de los documentos de entrada X representado como una bolsa de palabras X_i, de pertenecer a una lengua C_k y $p(X|C_k)$ es la tasa de la frecuencia de una palabra dentro de una categoría determinada.

$$p(X|C_k) = \prod_i p(X_i|C_k) \tag{5.10}$$

A continuación mediante la ecuación 5.11, Langdetect calcula la probabilidad de la clase o lengua C_k para maximizar *a posteriori*, donde $p(C_k)$ es el *a priori* de una categoría determinada.

$$p(C_k|X) = \frac{p(X|C_k)p(C_k)}{p(x)} \, \alpha \, p(C_k) \prod p(X_i|C_k) \tag{5.11}$$

Este módulo es opcional para el funcionamiento de DENISE ya que el usuario puede seleccionar manualmente la lengua de trabajo, sin embargo, se usó esta implementación de la librería Langdetect para trabajar automáticamente con documentos obtenidos desde la red sin necesidad de asignar una lengua de trabajo a cada fuente o grupo de fuentes a ser analizadas.

5.4 Modelos TF-IDF para la generación de representaciones de documentos

Después de haber asignado una lengua de trabajo el sistema procede a detectar la temática principal del texto de entrada, esta temática puede ser cualquiera de las mencionadas en la sección 5.2.1. Puesto que las unidades de interés de este proyecto se limitan a NS que se originan en el campo de la informática, tratamos la detección de tema como un problema de clasificación binaria donde las dos posibilidades son: *informática* o *no informática*, esta segunda categoría agrupa el resto de las temáticas.

Para llevar a cabo esta clasificación creamos representaciones vectoriales de los corpus especializados mediante la *frecuencia de término–frecuencia inversa de documento* (TF-IDF). La representación de cada clase (o tema) está compuesta por el conjunto de documentos que componen esta clase, tras haber procesado y normalizado cada texto. Las representaciones de cada temática sirven como un modelo de referencia para *detectar* a qué clase o tema puede pertenecer un texto de entrada. Este proceso en realidad asigna una clase *n* a un texto mediante un modelo de clasificación.

La *frecuencia de término* (TF), ecuación 5.12) fue definida por Luhn (1957) y consiste en calcular la frecuencia f máxima normalizada de ocurrencias de una palabra t dentro de un texto d.

$$\mathrm{TF}_{t,d} = \frac{f(t,d)}{max\{f(t,d) : t \in d\}} \tag{5.12}$$

Como complemento al algoritmo anterior, Sparck-Jones (1972) propuso la metodología *frecuencia inversa de documento* (IDF, ecuación 5.13), donde $|D|$ es el total de documentos de un corpus y $|\{d \in D : t \in d\}|$ la cantidad de documentos del corpus donde aparece t. Esta ecuación da mayor valor o peso a los términos menos frecuentes en una colección de textos, en vista de que estos pueden ser más informativos.

$$\mathrm{IDF}(t, D) = log\frac{|N|}{|\{d \in D : t \in d\}|} \tag{5.13}$$

Con las definiciones anteriores podríamos definir $\mathrm{TF} - \mathrm{IDF}$ como el producto de $\mathrm{TF}(t, d)$ e $\mathrm{IDF}(t, D)$, tal como se muestra en la ecuación 5.14.

$$\mathrm{TF} - \mathrm{IDF}(t, d, D) = \mathrm{TF}(t, d) \times \mathrm{IDF}(t, D) \tag{5.14}$$

La metodología TF-IDF (ecuación 5.14) ha sido extensamente utilizada en motores de búsqueda y sistemas de minería de datos (Salton et al., 1975, 1983; Salton y McGill, 1984; Salton y Buckley, 1988; Wu et al., 2008; Manning et al., 2008, 2009), puesto que permite calcular la importancia que tienen la distribución de palabras dentro de una colección de textos o corpus.

En nuestro caso, las representaciones creadas de cada documento nos permiten delimitar cada temática en función de sus palabras más relevantes. Los parámetros para generar nuestros modelos incluyen frecuencia de término mínima de 5 ocurrencias, normalización l_2[12] y frecuencia de término sublineal. Este último proceso consiste en añadir un peso a la frecuencia de término de la siguiente manera $wtf = 1 + log(tf)$, con la finalidad de evitar que los términos con mayor frecuencia sean los más relevantes.

$$l_2 = ||\mathbf{w}||_2^2 = \sum_{i=1}^{n} w_i^2 \tag{5.15}$$

Debido a la gran cantidad de dimensiones que tiene cada modelo empleamos la metodología t-SNE (Van Der Maaten y Hinton, 2008; Van Der Maaten, 2009; Van Der Maaten y Hinton, 2012; Van Der Maaten, 2014) para reducir la dimensionalidad de los modelos y de esta forma visualizar la distribución de los temas de cada lengua. Para generar las gráficas empleamos una muestra del 33 % de cada conjunto de datos. Podemos visualizar la distribución de los elementos que conforman nuestros modelos TF-IDF en la figuras 5.10 (catalán), 5.11 (español) y 5.12 (francés). Cada color representa una temática y cada punto representa un documento y su distribución en el espacio vectorial.

Mediante el uso de TF-IDF para crear representaciones de nuestros corpus especializados podemos obtener los atributos que delimitan cada temática. Su función principal es la de contener los datos de entrada entrenar un modelo de clasificación automática que servirá para asignar una temática a los textos que los usuarios desean analizar. En los apartados siguientes describiremos las metodologías de clasificación mediante aprendiza-

[12]La regularización o normalización l_2 (ecuación 5.15.) consiste en la suma de los cuadrados de todos los pesos w de los atributos, este proceso obliga a que los pesos sean pequeños –evitando que lleguen a cero– para prevenir el sobreajuste de los modelos generados.

je supervisado más comúnmente empleadas, así como los parámetros de entrenamiento empleados para generar nuestros modelos y una comparativa final.

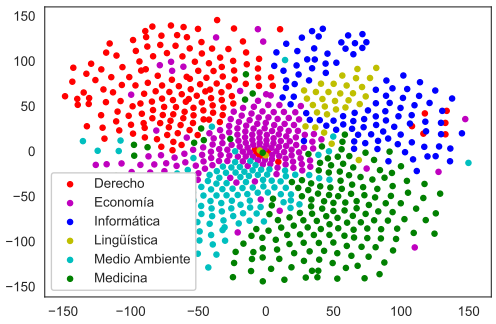


Figura 5.10 – Modelo TF-IDF para clasificación de temas en catalán.

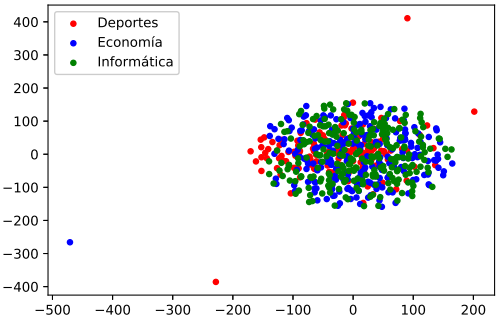


Figura 5.11 – Modelo TF-IDF para clasificación de temas en español.

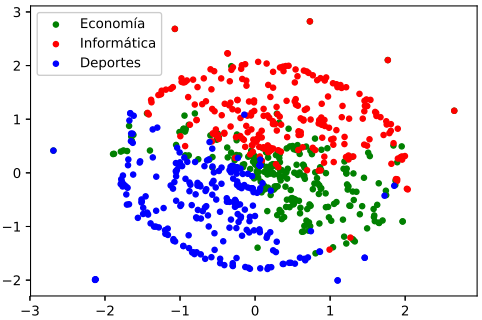


Figura 5.12 – Modelo TF-IDF para clasificación de temas en francés.

5.5 Clasificación mediante aprendizaje automático supervisado como estrategia para detección de temas

Dentro del campo del aprendizaje automático supervisado existen dos tipos de problemas principales: problemas de regresión y los problemas de clasificación. Para Hastie

et al. (2017), estas dos vertientes se encuentran en función del tipo de variables que un modelo puede predecir o generalizar. Un modelo de regresión tiene la capacidad de predecir salidas cuantitativas (probabilidades, índices, temperaturas, etc.), mientras que las variables cualitativas, también referidas como categóricas o discretas (especies de plantas, variedades taxonómicas, emociones, etc.) son las salidas esperadas de los modelos de clasificación.

Para comprender en qué consiste el proceso de clasificación y qué es un modelo de clasificación, se analizaron diferentes autores que abordan esta problemática desde perspectivas diferentes, pero relacionadas entre sí: análisis estadístico, aprendizaje automático y redes neuronales. Una de las definiciones clásicas es la propuesta por Michie et al. (1994), quienes expresan que la clasificación tiene como objetivo establecer una regla mediante la cual podemos asignar una clase (del repertorio de clases disponibles) a un elemento observado (Michie et al., 1994, p. 6).

En términos análogos, Duda et al. (2001) indican que la finalidad de un modelo de clasificación es dar una sugerencia de acción cuando el modelo es presentado con patrones o información novedosa (Duda et al., 2001, p. 8). Por otra parte, Bishop (2006), formaliza el concepto de clasificación como un conjunto de clases o etiquetas discretas, donde solo una clase del conjunto puede ser asignada a cada vector de entrada. Bajo este supuesto, el objetivo de la clasificación es tomar un vector de entrada x y asignar una de las K clases discretas C_k disponibles, donde $k = 1, \ldots, K$ (Bishop, 2006, p. 179). Las clases suelen ser distintas y separables, de forma que a cada entrada sea asignada una única clase. El espacio de las entradas se encuentra dividido en región de decisión cuyos límites son denominados *límites o superficies de decisión*.

De forma similar, Haykin (2009) habla de patrones de clasificación, es decir, cada elemento, señal o evento tiene un patrón característico que determina a qué clase —de un subconjunto de clases predeterminadas— puede pertenecer. Por ejemplo, una tarea de clasificación de patrones que requiere asignar distintas categorías (o clases) a una señal de entrada (Haykin, 2009, p. 3). En otros términos, la clasificación automática se refiere al proceso de llevar a cabo la predicción de una etiqueta para un punto no etiquetado (Zaki y Meira, 2014, p. 466), entendiendo *punto* como una abstracción de cualquier tipo de dato con atributos analizables.

En general todas las posturas mantienen la misma lógica expresada en diferentes términos: dado un conjunto de datos de entrada cuyos atributos determinan su pertenencia a una clase, un modelo de clasificación debe ser capaz de asignar de forma correcta la clase a la que dicho conjunto de datos pertenecen. Bajo este supuesto, existe un número finito de clases dentro del modelo y solamente la clase con mayor probabilidad es la que será asignada al conjunto de datos de entrada.

De acuerdo con Nilsson (1996), estos elementos de entrada pueden ser de diversos tipos, por ejemplo: "[...]the input vector is called by a variety of names. Some of these are: *input vector*, *pattern vector*, *feature vector*, *sample*, *example*, and *instance*. The components, x_i, of the input vector are variously called *features*, *attributes*, *input variables*, and *components*" (Nilsson, 1996, p. 9). En el caso de esta tesis, nuestros vectores de entrada serían los textos de entrada convertidos en representaciones vectoriales, donde cada palabra del texto representa un elemento del vector de entrada.

Así mismo, los datos de salida de los modelos de clasificación también pueden ser de distintos tipos: "[...]the output may be a categorical value, in which case the process

embodying h is variously called a *classifer*, a *recognizer*, or a *categorizer*, and the output itself is called a *label*, a *class*, a *category*, or a *decision*" (Nilsson, 1996, p. 9). Apegándonos a estas definiciones de datos de entrada y salida, nuestro modelo de clasificación debe de generar como salida un valor categórico. En este caso, cada uno de los temas que nuestro modelo puede predecir corresponde a una etiqueta o clase como valor esperado de salida que corresponde a una de las siguientes opciones: *informática*, *deportes* o *economía*.

La capacidad que tiene un modelo para predecir con exactitud cada una de las clases que han sido entrenadas se denomina *generalización*. Este concepto puede ser aplicado tanto a modelos de clasificación como a modelos neuronales u otros sistemas de aprendizaje automático. Bishop (2006) define este concepto como la habilidad de categorizar nuevos ejemplos que difieren de aquellos usados durante el entrenamiento (Bishop, 2006, p. 6), mientras que, Haykin (2009) considera que una red generaliza de forma adecuada cuando un modelo entrenado proyecta resultados correctos con elementos de entrada-salida a partir de un subconjunto de datos no empleados durante el proceso de entrenamiento de la red.

El nivel de generalización de un modelo depende de varios factores que se entrelazan: la calidad de los datos o el modelo empleado durante el entrenamiento, la calidad y el tipo de datos de entrada que se espera clasificar y el método de clasificación empleado ya que no existe un método que de una solución a todos los problemas. Existen tres tipos de modelos principales en materia de clasificación automática y cada acercamiento cuenta con métodos que se pueden adaptar a diferentes tipos de datos:

- Modelos generativos lineales: modelos bayesianos (gaussiano, multinomial, Bernoulli, etc.).
- Modelos discriminantes lineales: máquina de vectores de soporte, regresión logística y perceptrón multicapa.
- Modelos en conjunto: árboles de decisión y bosques aleatorios.

En la sección anterior se habló sobre la metodología TF-IDF y se explicó el proceso que se siguió para entrenar un modelo TF-IDF empleando cada corpus especializado. Estas representaciones, en conjunto con un modelo de clasificación automática, funcionan como un instrumento para signar un tema dentro de nuestro sistema. Así, un documento de entrada se compara frente a un modelo de clasificación de tema y el modelo asigna una temática en función de la similitud que tienen los atributos de dicho texto de entrada frente a cada una de las categorías disponibles.

Empleamos el enfoque de la clasificación automática de documentos como metodología para la detección de temas y, desde esta perspectiva, entendemos *documento* como cualquier texto que puede ser analizado —indistintamente de su extensión— de forma tal que sea posible determinar su temática en función de su contenido textual.

En los siguientes apartados se describen los principales métodos de cada uno de los tres tipos de modelos mencionados anteriormente: clasificador bayesiano multinomial, máquina de vectores de soporte, regresión logística, perceptrón multicapa y bosques aleatorios. La finalidad de esta descripción es comprender las ventajas y desventajas de cada acercamiento para seleccionar un modelo que, al mismo que sea capaz de generalizar

adecuadamente, el tiempo de entrenamiento y el consumo de recursos sean eficientes y viables para ser implementados en nuestro sistema propuesto.

5.5.1 Regresión logística

A pesar de su nombre, la regresión logística (Cox, 1958; Walker y Duncan, 1967) es un método de clasificación que nos permite calcular la probabilidad de un elemento de pertenecer a una clase determinada con respecto a la totalidad de clases existentes en el modelo. La regresión logística intenta modelar la relación que existe entre $p(X) = Pr(Y = 1|X)$ y X, tomando como punto de partida un modelo de regresión lineal, tal como se muestra en la ecuación 5.16.

$$p(X) = Pr(Y = 1|X) \tag{5.16}$$

Las probabilidades de predicción son modeladas mediante la función logística (ecuación 5.17), originalmente definida por Verhulst (1845). Se emplea dicha función para modelar $p(X)$ de forma tal que dé resultados entre 0 y 1 para cada valor de X.

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \tag{5.17}$$

Para ajustar el modelo se suele emplear una metodología conocida como máxima verosimilitud (*maximum likelihood* en inglés). Esta función (ver ecuación 5.18) busca estimar los valores de los coeficientes de peso β_0 y β_1 en la ecuación 5.17, de forma que $p(X)$ de como resultado un valor entre 0 y 1.

$$\ell(\beta_0, \beta_1) = \prod_{i:y=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x'_i)) \tag{5.18}$$

Después de despejar 5.17 obtenemos 5.19, donde $p(X)/(1-p(X))$ son probabilidades que pueden tener valores entre 0 e ∞.

$$\frac{p(X)}{1 - p(X)} = e^{\beta_0+\beta_1 X} \tag{5.19}$$

Finalmente, despejando el logaritmo de la función obtenemos la ecuación 5.20, donde un incremento en X cambia las probabilidades por β_1.

$$\log\left(\frac{p(X)}{1 - P(X)}\right) = \beta_0 + \beta_1 X \tag{5.20}$$

La regresión logística se clasifica dentro de los modelos lineales. Este tipo de modelos pueden ser implementados para realizar predicciones, tanto con modelos de clasificación binaria (ecuación 5.21), como para modelos de clasificación multiclase (ecuación 5.22). En el primer caso, el resultado esperado de $\hat{y}$ con los pesos w, $x[n]$ atributos de punto de datos y la intersección b es una función lineal (hiperplano), cuyo resultado esperado es

una función que delimita dos clases distintas.

$$\hat{y} = w[0] \times x[0] + w[1] \times x[1] + \ldots + w[p] \times x[p] + b > 0 \tag{5.21}$$

El segundo caso, la clasificación multiclase, se utiliza para realizar predicciones de múltiples clases que pueden ser delimitadas empleando la metodología *1 contra el resto* (*1 vs. rest*), donde el clasificador se entrena para diferenciar cada una de las clases en contra del conjunto de clases restantes. En el caso de la regresión logística esta estrategia para entrenar clasificadores multiclase, no es la única que puede ser implementada ya que existen otras metodologías que siguen la misma lógica: un vector con coeficientes w y una intersección b por clase.

$$\hat{y} = w[0] \times x[0] + w[1] \times x[1] + \ldots + w[p] \times x[p] + b \tag{5.22}$$

La regresión logística calcula las probabilidades posteriores de las clases K mediante funciones lineales en x, mientras que, al mismo tiempo, se asegura de que la suma de dichas probabilidades tenga un resultado igual a 1 y las probabilidades se mantengan dentro de un rango [0,1] (Hastie et al., 2017). Nuestro modelo de regresión logística implementa un modelado para clasificación multiclase en conjunto con una norma de regularización L2, de forma que el objetivo es minimizar una función de costo.

Entrenamos los modelos de regresión logística multiclase mediante un enfoque multinomial. Seleccionamos un valor del inverso de fuerza de regularización *C* igual 1, un solucionador LBFGS (Liu y Nocedal, 1989) (*limited memory Broyden – Fletcher – Goldfarb – Shanno algorythm*[13]) y 0.00001 de tolerancia en 100 iteraciones.

5.5.2 Máquinas lineales de vectores de soporte

Las máquinas de vectores de soporte (SVM) son un algoritmo que genera representaciones de los vectores de entrada x en un espacio de atributos multidimensionales Z mediante una representación no lineal, seleccionada priori. En este espacio (ver figura 5.13) un hiperplano separador es construido (Vapnik, 1995, pp. 133-134), dicho espacio es la delimitación entre los puntos marginales (vectores de soporte) de cada clase. A su vez, son estos puntos los que soportan de manera óptima las diferencias entre clases.

La metodología *1-vs-rest*, tal como es implementada por Vapnik (1999), consiste en construir K SVM diferentes, en los cuales el modelo con índice K es entrenado usando los atributos de las clases C_K como ejemplos positivos y los atributos de las clases $K-1$ restantes, como los ejemplos negativos (Bishop, 2006, p. 338), podría describirse como múltiples clasificadores lineales analizando cada uno un conjunto de datos que conforman un grupo frente el resto de los conjuntos agrupados como uno único.

En palabras de Boser et al. (1992), la implementación multiclase de los SVM permite utilizar núcleos o *kernels* para solucionar problemas de clasificación no lineal, en este caso utilizamos un *kernel* lineal implementado mediante la librería LIBLINEAR (Fan et al., 2008), puesto que scikit-learn implementa dicha librería. Nuestro modelo de clasificación

[13] Versión de memoria limitada del algoritmo Broyden–Fletcher–Goldfarb–Shanno (Fletcher, 1987), que se adapta a datos que tienen un gran número de variables, por ejemplo corpus textuales.

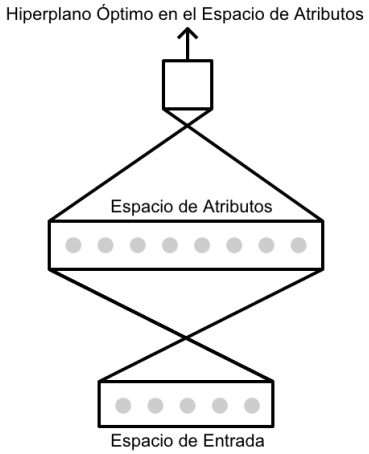


Figura 5.13 – Proceso de generación de un hiperplano óptimo (Vapnik, 1995, pp. 134).

mediante SVM fue entrenado con los siguientes parámetros: penalización L2, pérdida *squared hinge*[14] (ver ecuación 5.23.), tolerancia igual a 0.0001, un valor de regularización C igual a 1 y con metodología *1-vs-rest* en un máximo de 100 iteraciones.

$$L(y, \hat{y}) = \sum_{i=0}^{N} (\text{máx}(0, 1 - y_i \cdot \hat{y}_i)^2) \tag{5.23}$$

5.5.3 Clasificador bayesiano ingenuo multinomial

Este método de clasificación probabilística se fundamente en el teorema de Bayes 5.24, pero implementado bajo el principio de que existe independencia entre cada par de parámetros relacionados con los valores de las variables de clase. Las clases (o temáticas en el caso de esta tesis) más frecuentes tienen mayor probabilidad de ser la clase correcta.

$$P(y|x_1, \ldots x_n) = \frac{P(y)P(x_1, \ldots x_n|y)}{P(x_1, \ldots x_n)} \tag{5.24}$$

Cuando se trata de clasificación de documentos, este algoritmo, suele ser el clasificador bayesiano ingenuo multinomial (ver 5.25) donde cada parámetro condicional $\hat{P}(t_k|c)$ es un peso que indica qué tan buen indicador para c es un elemento t_k. De forma similar, el $log\hat{P}(c)$ priori es el peso que indica la frecuencia relativa de c (Manning et al., 2008, pp. 258–260). La suma del log priori y los pesos de los términos es la magnitud que indica la cantidad de evidencia que existe para determinar que un documento pertenece a una clase en concreto.

$$c_{map} = \underset{c \in \mathbb{C}}{argmax} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right] \tag{5.25}$$

[14]Donde $\hat{y}$ es el valor se espera predecir e y tiene un valor de 1 o -1.

Nuestro modelo fue entrenado usando un valor de α igual a 1, es decir, implementamos el suavizado de Laplace tras aplicar la regla de la cadena. Ajustamos el priori con un valor de verdadero y el priori de la clase con un valor nulo. La diferencia fundamental que existe entre los clasificadores bayesianos radica en los supuestos que llevan a cabo con relación a la distribución de $P(x_i, y)$. El suavizado de Laplace sirve para eliminar los ceros, sumando 1 a cada instancia, tal como se expresa en la ecuación 5.26, donde, en este caso, $B = |V|$ es el número de unidades en un vocabulario.

$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \in V}(T_{ct'}) + 1} = \frac{T_{ct} + 1}{(\sum_{t' \in V(T_{ct'})}) + B'} \tag{5.26}$$

Los modelos bayesianos son extensamente empleados en el campo del procesamiento del lenguaje natural (PLN) (Manning et al., 2008; Jurafsky y Martin, 2009), por ejemplo como algoritmo base para realizar clasificación de documentos, detección de spam, análisis de sentimiento, o como método de referencia para realizar evaluaciones posteriores con modelos de mayor complejidad.

5.5.4 Clasificador de bosques aleatorios

Los bosques aleatorios son un método de clasificación por conjunto (Opitz y Maclin, 1999; Polikar, 2006) ya que implementan una serie de métodos (en nuestro caso los árboles de decisión) agrupados mediante un algoritmo de aprendizaje, para mejorar los resultados de salida promediando os resultados de los valores obtenidos por cada componente del conjunto.

En términos simples, un bosque aleatorio implica el uso de múltiples árboles de decisión (Müller y Guido, 2017, p. 83), donde cada árbol analiza una parte de los datos y posteriormente el conjunto de árboles (bosque) da como resultado el modelo de clasificación final. En la figura 5.14 podemos ver este proceso de forma simple: cada árbol analiza un grupo de parámetros y en conjunto "votan" para obtener una predicción.

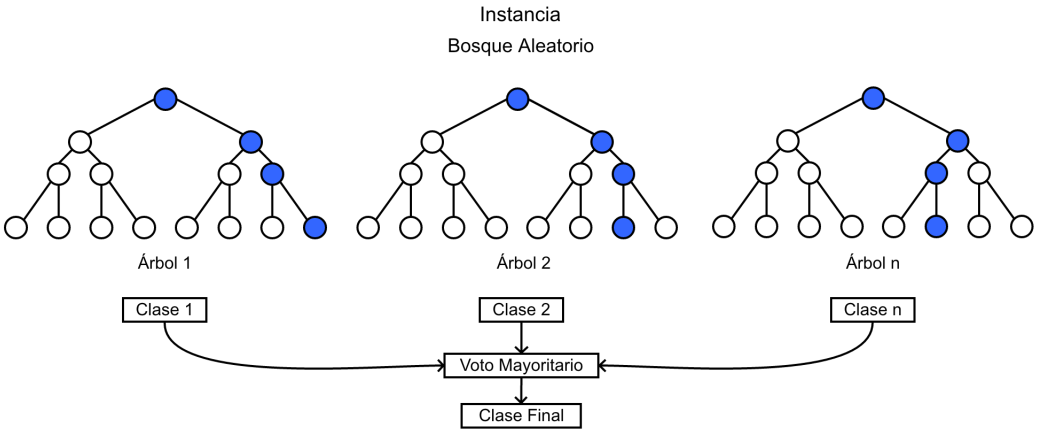


Figura 5.14 – Diagrama de bosque aleatorio.

A diferencia del *bagging*[15] (Breiman, 1996), los bosques aleatorios no consideran la correlación entre cada árbol de decisión, sino que cada árbol es independiente, no obstante

[15] Acrónimo de *bootstrap aggregation*.

ambos métodos involucran el uso de múltiples árboles de decisión en paralelo con la finalidad de reducir la varianza.

Está metodología, originalmente planteada por Ho (1995, 1998), emplea reglas de clasificación estocástica (Kleinberg, 1990, 1996) que consisten en tomar como entrada soluciones parciales o pobres y usarlas en conjunto para crear soluciones más adecuadas. Nuestra implementación se basa en la propuesta de Breiman (2001) en conjunto con el algoritmo de impureza[16] de Gini (ecuación 5.27). Este índice determina la pureza de una partición $\mathbf{D}$ de un árbol, cuando la mayoría de probabilidades de una clase es 1, el resto de las clases es 0. Por otra parte, cuando cada clase se encuentra representada por una probabilidad $P(c_i|\mathbf{D}) = \frac{1}{k}$ el índice de Gini tiene un valor de $\frac{k-1}{k}$.

$$G(\mathbf{D}) = 1 - \sum_{i=1}^{k} P(c_i|\mathbf{D}^2) \tag{5.27}$$

Como parámetros de entrenamiento de nuestro modelo usamos 200 estimadores, es decir, 200 árboles de decisión, con una profundidad de 3 niveles, donde cada nivel de profundidad representa un nodo de cada cada árbol de decisión.

5.5.5 Perceptrón multicapa

El perceptrón multicapa (MLP), también conocido en inglés como *deep feedforward network*[17], *feedforward neural network* o *multilayer perceptron*, es un tipo de red neuronal compuesta por una serie de neuronas o perceptrones (McCulloch y Pitts, 1943; Rosenblatt, 1958, 1962; Rumelhart et al., 1986; Widrow y Lehr, 1993) que, en conjunto, son capaces de resolver problemas de clasificación no lineal a deferencia de un único perceptrón (Minsky y Papert, 1988; Bishop, 1995; Gurney, 1997; Haykin, 2009).

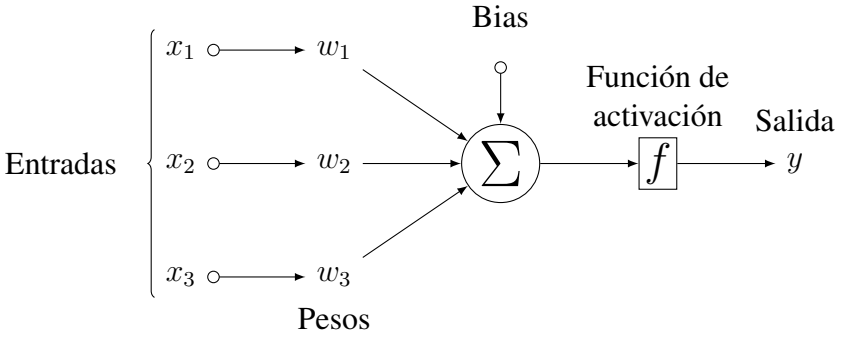


Figura 5.15 – Diagrama de un perceptrón o neurona.

Los elementos esenciales de un perceptrón (ver figura 5.15) son un vector de entrada X con atributos $(x_i, \ldots, x_n)$, un vector de pesos[18] representados por el vector W $=$

[16]La *pureza* cuantifica en qué medida un agrupamiento contiene entidades de una misma clase, es decir, qué tan "puro" es cada árbol (Zaki y Meira, 2014, p. 426).

[17]El término *feedforward* se refiere a que los datos de entrada fluyen desde x, a través de los cálculos intermedios usados para definir f y finalmente hacia la salida y sin conexiones de retroalimentación (Goodfellow et al., 2016, p. 168).

[18]Entendemos *peso* como un número real que representa la conectividad entre dos nodos.

$(w_1, \ldots, w_n)$, una función de activación (o umbral) y una función de salida f definidas en la ecuación 5.28. El perceptrón tiene una salida igual a 1 cuando $\sum_{n=1}^{n} x_i w_i \geq \theta$, mientras que en otro caso la salida será igual a 0. Cuando el umbral θ se establece en 0, los valores arbitrarios de los umbrales se asignan usando vectores Y y V con una dimensionalidad de $(n + 1)$ cuyos primeros componentes son los mismos de los vectores X y W. Esta estructura suma las entradas con sus pesos y compara esta suma con el valor del umbral para obtener un resultado o salida (Nilsson, 1996).

$$f = \text{activación}(\sum_{i=1}^{n+1} w_i x_i, 0) \tag{5.28}$$

El rol de las funciones de activación es determinar el nivel de respuesta de una neurona frente a un estímulo. Un ejemplo sencillo sería pensar en los resultados como valores binarios, donde 0 indica que una neurona no se encuentra activada y 1 que ha sido activada. Usar funciones de activación lineales tiene los siguientes inconvenientes: no permiten que las redes tomen datos complejos como entrada ni el uso de la retropropagación como método de entrenamiento. No obstante, se puede transformar la combinación de valores lineales (suma de atributos y pesos) mediante una función no lineal (Nilsson, 1996), de forma que dichas funciones no lineales permitan el uso de la retropropagación.

Tradicionalmente se han usado funciones de activación sigmoides, mientras que en nuestra implementación usamos la unidad lineal rectificada (en inglés *rectified linear units* o ReLU, ver ecuación 5.29). El rectificador ReLU toma como entrada un valor x_i y da como resultado 0 cuando $x < 0$ y una función lineal cuando $x \geq 0$ (Nair y Hinton, 2010; Glorot y Bengio, 2010; Glorot et al., 2011; Agarap, 2018).

$$f(x_i) = max(0, x_i) f(x) = \begin{cases} 0 & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases} \tag{5.29}$$

Tal como se muestra en la figura 5.16, la estructura básica de este tipo de redes neuronales se compone de, como mínimo, tres capas: la capa de entrada que contiene los atributos y sus pesos, una capa oculta donde ocurre el aprendizaje mediante una función de activación (ReLU en nuestro caso) y una capa de salida donde se obtienen los resultados del aprendizaje. Esta arquitectura de redes neuronales es la base de paradigmas modernos como el aprendizaje profundo (Schmidhuber, 2015; Goodfellow et al., 2016; Deng y Liu, 2018).

Torres-Moreno (1997) define el aprendizaje como el proceso de adaptación de los parámetros de un sistema para obtener una respuesta deseada frente a una entrada o un estímulo. Cuando dicha respuesta no es la esperada se deben reajustar los pesos de la red, de forma que se minimice el error general de la red. Esta técnica para el aprendizaje se conoce como retropropagación de errores (Rumelhart et al., 1986) y consiste en calcular el descenso del gradiente[19] (Hadamard, 1908) de las diferentes variables y sus pesos, en relación con los resultados esperados durante el entrenamiento para minimizar el error $\mathcal{E}$. Este proceso actualiza los pesos de las neuronas (aumentando o disminuyendo su valor)

[19]El término *descenso del gradiente* suele hacer referencia al *descenso del gradiente estocástico* (Robbins y Monro, 1951; Kiefer y Wolfowitz, 1952; Chung, 1954; Kantorovich y Akilov, 1982; Barzilai y Borwein, 1988; Bottou et al., 2016).

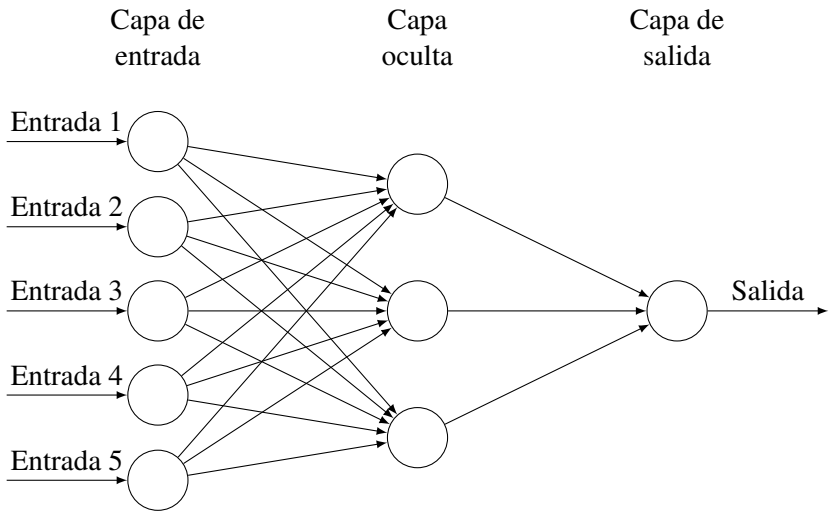


Figura 5.16 – Diagrama de un perceptrón multicapa.

para encontrar un mínimo local óptimo. Entenderemos *error* en términos de la diferencia que existe en los datos de entrada, pesos de cada capa oculta y la salida esperada.

El proceso de retropropagación se lleva a cabo en cada neurona de la red comenzando desde la penúltima capa oculta en dirección de la capa de entrada[20]. El resultado de este proceso indica en qué medida cada neurona debe ajustar su peso para mejorar las conexiones entre nodos, para reducir el error total de la red. Así, dada una respuesta deseada d_i de un vector de entrada $\mathrm{X_i}$ en el conjunto de entrenamiento Ξ, se calcula el error cuadrado del conjunto de entrenamiento mediante la ecuación 5.30, donde f_i es la salida esperada del vector de entrada $\mathrm{X_i}$. Finalmente, el valor de cada peso de la red es proporcional al negativo de la derivada parcial del error $\mathcal{E}$ de cada peso.

$$\mathcal{E} = \sum_{\mathrm{X_i} \in \Xi (d_i - f_i)^2} \tag{5.30}$$

Para calcular el descenso del gradiente, nuestra implementación usa el optimizador Adam (*Adaptive Moment Estimation* en inglés) (Kingma y Ba, 2014). Adam es un algoritmo de optimización estocástica que está basado en los algoritmos AdaGrad (Duchi et al., 2011) y RSMprop (Tieleman y Hinton, 2012). Adam combina las ventajas de AdaGrad y RSMproop: funciona de manera óptima con datos dispersos[21], tanto en línea (ejemplo a ejemplo) como en lotes (conjunto de aprendizaje completo). La optimización estocástica es el proceso fundamental de aprendizaje de las redes neuronales que utilizan el descenso de gradientes para minimizar el error encontrando un valor mínimo local.

En nuestro caso, los atributos de entrada del MLP son las representaciones TF-IDF de los corpus especializados de cada lengua y las salidas esperadas las temáticas contenidas en cada corpus. Entrenamos nuestro MLP con 100 neuronas, usando la función de

[20]De este hecho parte la idea de la retropropagación, las redes neuronales son alimentadas hacia "adelante" y corrigen sus pesos desde "atrás".

[21]Por ejemplo, una matriz de grandes dimensiones cuyos valores son mayoritariamente 0.

activación ReLU con un optimizador Adam en un máximo de 200 iteraciones. Como parámetros específicos seleccionamos un valor del hiperparámetro α = 0.0001 (regularización L2), tolerancia de optimización 1e-4, momento igual 0.9 y entrenamos el algoritmo en mini lotes de tamaños iguales calculados mediante $lote = \min(200, n_{muestras})$ con una tasa de aprendizaje constante. Como parámetros específicos del optimizador Adam seleccionamos un tasa de deterioro exponencial del primer vector de momento β_1 = 0.9, tasa de deterioro del segundo vector de momento β_2 = 0.99 y un valor de estabilidad numérica ϵ = 1e-08

5.5.6 Comparación de modelos de clasificación temáticos

Puesto que nuestro conjunto de datos se trata de modelos de clasificación multiclase, cada palabra de cada modelo representa un atributo y cada categoría o tema representa una clase, empleamos la exactitud como métrica de validación de resultados. En catalán (ver figura 5.17), los modelos SVC lineal (SVC), regresión logística (LR) y perceptrón multiacapa (MLP) obtuvieron resultados similares en exactitud (E): SVC, E = 0.93; LR E = 0.91; y MLP E = 0.92. En último término encontramos el clasificador de bosque aleatorio (RFC) con E = 0.74, este fue el valor más bajo observado de los modelos que fueron evaluados.

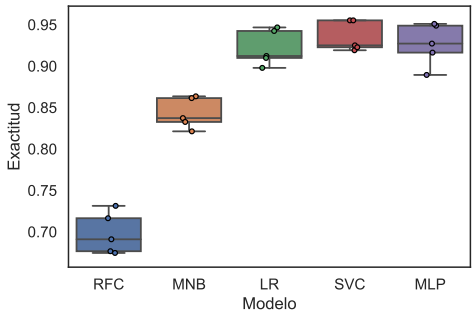


Figura 5.17 – Comparación de modelos para predicción de tema en catalán.

En español (5.18) observamos resultados similares, pero con un mayor porcentaje de exactitud que en catalán: SVC, E = 0.98; LR, E = 0.98; y MLP E = 0.98. Estos resultados con el conjunto de prueba pueden indicar que este modelo generalizará mejor que el modelo catalán y el modelo francés. En último término encontramos de nuevo al RFC con E = 0.79.

En francés (5.19) se repite el mismo patrón: E = 0.92 el SVC; LR, E = 0.92 ; MLP, E = 0.92; y RFC, E = 0.80. Podemos decir que, para esta tarea en concreto, el uso de un modelo de bosque aleatorio sería no aconsejable. Posiblemente la causa de su bajo desempeño se deba a la gran cantidad de etiquetas que deben predecirse.

En la tabla 5.8 podemos ver un resumen de la exactitud de cada modelo en cada lengua, los modelos que obtuvieron mejores resultados en todas las lenguas fueron el clasificador por regresión logística, SVC lineal y el MLP, no obstante, el tiempo de entrenamiento es sustancial, sobre todo entre el modelo de regresión logística y el MLP. Los modelos más complejos han llevado más tiempo de entrenamiento en comparación con los modelos más simples y por lo tanto también implican el uso de mayor tiempo de carga, predicción y consumo de recursos.

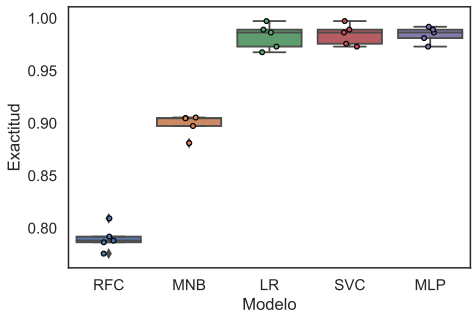


Figura 5.18 – Comparativa de modelos para predicción de tema en español.

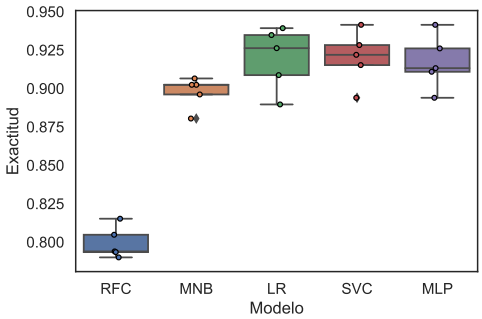


Figura 5.19 – Comparativa de modelos para predicción de tema en francés.

En la sección 5.6 analizaremos en profundidad los resultados de los tres modelos que, en cada lengua, obtuvieron mejores resultados: MLP, SVC lineal y regresión logística. A pesar de que los resultados de la evaluación de exactitud nos muestran que los tres modelos cuentan con la capacidad de generalizar adecuadamente, resta realizar pruebas con las concordancias que contienen NS y evaluar con otras métricas como precisión, exhaustividad y *f1-score*. También, para seleccionar un modelo definitivo, debemos tener en cuenta las ventajas o desventajas de utilizar un modelo más complejo, o más sencillo, cuando ambos pueden generalizar de forma similar.

5.6 Extracción de palabras claves

El siguiente paso de nuestra metodología consiste en la extracción de las palabras claves que se encuentran dentro del texto de entrada. Existen diversos métodos para llevar a cabo este proceso, por ejemplo: análisis de n-gramas, RAKE, TAKE, TextRank (Rose et al., 2010; Li y Wang, 2014; Pay, 2016; Pay y Lucci, 2017; Pay et al., 2018) e implementaciones de estos métodos en conjunto con medidas de similitud o fuentes externas de información.

Decidimos utilizar TextRank ya que este algoritmo no depende de fuentes externas de conocimiento (diccionarios, listas, bases de datos, etc.), sino que utiliza el contexto de aparición de cada palabra usada en un texto para determinar cuáles son las más relevantes. TextRank está inspirado en la metodología PageRank (Page et al., 1999; Richardson y Domingos, 2002) que consiste en una implementación de matrices de probabilidad de Markov (Strang, 2006) y grafos, pero adaptado al contenido textual y no solo al recorrido

Modelo	Exactitud Español	Catalán	Francés
SVC Linear (SVC)	0.9842	0.9323	0.9231
Regresión Logística (LR)	0.9826	0.9158	0.9231
Perceptrón Multicapa (MLP)	0.9842	0.9243	0.9227
Clasificador Bayesiano Multinomial (MNB)	0.8986	0.8204	0.8962
Bosque Aleatorio (RFC)	0.7902	0.7402	0.8016

Tabla 5.8 – Resumen de resultados de evaluación de exactitud de cada modelo de clasificación.

de un grafo.

Nuestra implementación de TextRank añade un filtro de etiquetas gramaticales al proceso original ya que, tal como confirmamos en la base de datos del OBNEO, la gran mayoría de los NS se concentran en tres categorías gramaticales: verbos, sustantivos y adjetivos. El filtro consiste en analizar todas las palabras del texto y sus relaciones sintácticas, pero solamente conservamos las categorías que son de interés. De esta forma tomamos en cuenta toda la información que nos puede aportar la concordancia de cada palabra, pero solamente extraemos aquellas palabras que pertenecen a las categorías gramaticales más productivas.

5.6.1 TextRank para extracción de palabras claves

El algoritmo original de TextRank propuesto por Mihalcea y Tarau (2004)[22] consiste en analizar un texto para obtener las palabras más relevantes dentro del contexto donde aparecen para construir un grafo con todas las asociaciones de palabras que existen en el texto. El algoritmo identifica cada palabra como un vértice del grafo. Posteriormente se crean nodos entre cada vértice que pueden tener pesos o no, una vez generados estos nodos el algoritmo itera hasta llegar a la convergencia.

PageRank (ver ecuación 5.31) toma como base el algoritmo PageRank (Page et al., 1999), donde $G = (V, E)$ es un grafo dirigido con un conjunto de vértices (o nodos) V y aristas E y, al mismo tiempo, E es un subconjunto de $V \times V$. Entendemos $In(V_i)$ como un conjunto de vértices que apunta a V_i como su predecesor, mientras que $Out(V_i)$ es el conjunto de vértices al cual el nodo V_i apunta como sucesores.

$$S(V_i) = (1 - d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out\ (V_j)|} S(V_j) \tag{5.31}$$

La variable d es un factor de ajuste que oscila entre 0 y 1, como convención d suele establecerse en 0.85. Dicho valor indica la probabilidad de saltar de un nodo hacia otro nodo aleatorio en el grafo. El algoritmo TextRank (ver ecuación 5.32) construye sobre PageRank, pero añade pesos w a las aristas para calcular el puntaje asociado a un nodo del grafo final.

[22] Las mejoras e implementaciones de este algoritmo se encuentran en Mihalcea (2004, 2005).

$$WS(V_i) = (1-d) + d * \sum_{V_j \in (V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \tag{5.32}$$

El algoritmo converge cuando la tasa de error de cualquier vértice del grafo cae por debajo de un umbral establecido (0.0001 según Mihalcea y Tarau (2004)). Mediante esta metodología se da prioridad a los nodos que tienen un mayor número de conexiones dentro del grafo (palabras que ocurren en conjunto con otras) y menos prioridad a aquellos nodos con menor coocurrencia. Mediante esta metodología conservamos las unidades que consideraríamos relevantes en el discurso, pero que no pueden ser extraídas mediante listas de exclusión, como es el caso de los NS. De esta forma, podemos generar una lista de palabras claves que posteriormente serán evaluadas frente al modelo de desambiguación y el modelo de clasificación para determinar si existe algún candidato a NS.

5.6.2 TextRank con filtro de etiquetas gramaticales

Nuestra implementación de TextRank toma como referencia el diseño de Barrios et al. (2016) y analiza los textos de entrada mediante la siguiente metodología: realiza un ciclo de preprocesamiento para obtener un texto limpio, posteriormente genera una matriz que incluye tokens, lemas y etiquetas de cada unidad del texto. Esta matriz se emplea para llevar a cabo los siguientes procesos: analizar las etiquetas que son de interés (verbos, sustantivos y adjetivos), incrementar una *stoplist* con las unidades que no son de interés e iniciar el análisis recursivo con TextRank.

El primer paso para realizar la extracción de palabras claves es generar una matriz en blanco de una dimensión igual al número de elementos generados durante el procesamiento. Posteriormente se genera un grafo (ver figura 5.20) que muestre las relaciones que existen entre cada palabra y se ejecuta TextRank con los siguientes parámetros: un máximo de 50 iteraciones, umbral de convergencia 0.0001, un factor de amortiguamiento 0.85 (similar a Mihalcea y Tarau (2004)) y una ventana de análisis de 5 palabras.

Este algoritmo, además, añade la opción de extracción por n-gramas, es decir, puede extraer unidades unilexemáticas o polilexemáticas con una extensión seleccionada a medida y limitar la cantidad de palabras que extraemos en función de la extensión del texto. Como prueba de funcionamiento usamos el mismo texto que se incluye en la bibliografía y empleamos la API del traductor de Google[23] para generar versiones en catalán, español y francés.

> Compatibility of systems of linear constraints over the set of natural numbers. Criteria of compatibility of a system of linear Diophantine equations, strict inequations, and nonstrict inequations are considered. Upper bounds for components of a minimal set of solutions and algorithms of construction of minimal generating sets of solutions for all types of systems are given. These criteria and the corresponding algorithms for constructing a minimal supporting set of solutions can be used in solving all the considered types of systems and systems of mixed types (Mihalcea y Tarau, 2004, p. 4).

[23] `https://cloud.google.com/translate/docs/?hl=es-419`

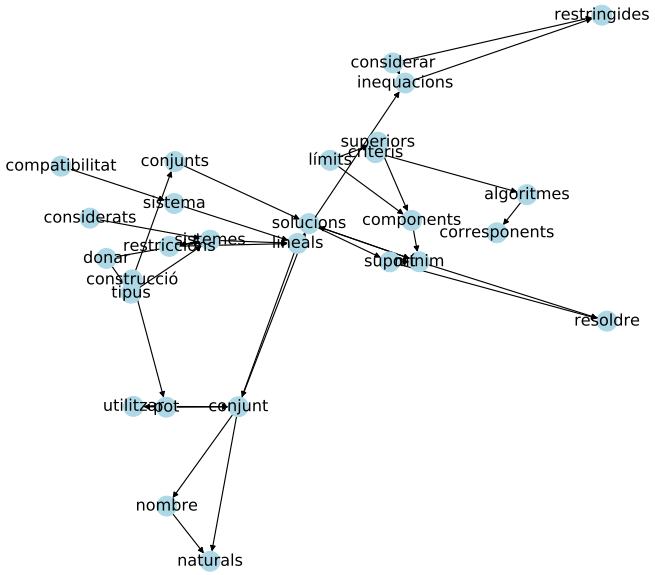


Figura 5.20 – Ejemplo de grafo generado con TextRank en catalán.

En las Tablas 5.9 y 5.10 se pueden observar los resultados obtenidos con esta metodología. Cabe señalar que las unidades polilexemáticas tienen puntuaciones más altas debido a que los elementos que conforman dichas unidades cuentan con un mayor peso. Es decir, al analizar recursivamente estas unidades se toman en cuenta las relaciones que tiene cada elemento con el resto de los componentes del texto y, al mismo tiempo, con los elementos que conforman la unidad polilexemática.

Palabras clave (Inglés)	Puntuación	Palabras clave (Español)	Puntuación
minimal supporting set	4.608	conjunto mínimo	3.338
minimal set	3.885	ecuaciones diofánticas lineales	2.959
minimal generating sets	3.402	conjuntos generadores mínimos	2.218
linear diophantine equations	2.929	inecuaciones estrictas	2.176
considered types	2.622	apoyo mínimo	2.089
corresponding algorithms	2.071	algoritmos correspondientes	2.042
linear constraints	2.053	restricciones lineales	2.022
nonstrict inequations	2.034	soluciones	1.938
strict inequations	1.992	considerados tipos	1.853
solutions	1.920	límites superiores	1.609

Tabla 5.9 – Palabras clave obtenidas con TextRank en inglés y español.

El algoritmo permite extraer palabras claves de forma no supervisada. Esta particularidad es relevante para nuestro sistema ya que es un proceso que se ejecuta en función de cada texto ingresado. En los ejemplos generados mediante traducción automática podemos encontrar algunas equivalencias entre palabras claves como *conjunto mínimo*, *minimal set*, *ensamble minimal* y *conjunt mínim*.

Nuestro enfoque se limitará a la extracción de unigramas o unidades unilexemáticas ya que la gran mayoría de los NS registrados en la base de datos son unidades de este

tipo. En el capítulo 6 presentamos una evaluación en profundidad de esta metodología, emplearemos los contextos de las tablas del OBNEO puesto que cada contexto contiene un NS y, por lo tanto, esperamos como resultado obtener la mayor cantidad de NS posibles. Como mencionamos en el apartado 5.2.2, contamos con tablas en catalán y español, sin embargo, no contamos con una base de datos equivalente en francés.

Palabras clave (Francés)	Puntuación	Palabras clave (Catalán)	Puntuación
ensemble minimal	3.268	conjunt mínim	3.468
équations diophantiennes linéaires	2.818	solucions mínimes	2.820
inéquations strictes	2.328	consideren inequacions	2.206
groupes générateurs minimaux	2.214	inequacions estrictes	2.203
algorithmes correspondants	2.029	suport mínim	2.170
contraintes linéaires	1.943	restriccions lineals	2.066
solutions	1.932	donen tipus	2.029
limites supérieures	1.564	considerats tipus	1.940
nombres naturels	1.522	límits superiors	1.637
systèmes	1.512	sistemes	1.569

Tabla 5.10 – Palabras clave obtenidas con TextRank en francés y catalán.

5.7 Métodos de aprendizaje profundo para la detección de neologismos semánticos

En la actualidad el concepto de aprendizaje profundo (*deep learning*) ha ganado bastante popularidad, puesto que ha sido un paradigma revolucionario tanto en el campo de la inteligencia artificial como en el del aprendizaje automático. En la sección 5.5.5 mencionamos que el perceptrón multicapa (MLP) es la base del aprendizaje profundo, por lo tanto, para comprender este concepto y su relevancia para el desarrollo de herramientas para el procesamiento del lenguaje natural, presentaremos conceptos clave de forma cronológica.

Uno de los primeros usos del término *profundo* y de la implementación de múltiples capas de entrenamiento se puede encontrar en Hinton et al. (2006). Los autores proponen un método de entrenamiento *profundo* que consiste en la implementación de dos (o más) capas de entrenamiento, en conjunto con previos complementarios (*complementary priors*) que tiene la capacidad de aprender redes de creencia dirigida. Así, mediante esta arquitectura se deriva un algoritmo más veloz que puede aprender redes de creencia profundas una capa a la vez. Este modelo generativo produce mejores resultados en clasificación de dígitos en comparación con otros algoritmos de aprendizaje discriminativo.

Ese mismo año, Hinton y Salakhutdinov (2006) presentan una metodología para inicializar los pesos de una red que permite que *autoencoder* profundo, aprenda mediante retropropagación de errores derivativos. Además, mencionan una metodología de iniciación de pesos que permite el aprendizaje de códigos de bajas dimensiones, esta configuración permite la reducción de las dimensiones totales de los datos. Finalmente destacan que, gracias a los avances tecnológicos de las últimas décadas, se tienen las condiciones necesarias para la implementación de modelos de aprendizaje profundo. Desde la década

de los 80 se tenía la intuición de que implementar *autoencoders* profundos con retropropagación de errores sería una técnica efectiva para la reducción no lineal, siempre y cuando se contara con el poder de procesamiento suficiente, los conjuntos de datos fueran extensos en tamaño y los pesos iniciales se encontraran cerca de una solución óptima. En la actualidad las tres condiciones se ven satisfechas.

Tres años más tarde, Salakhutdinov y Hinton (2009) muestran que el uso de múltiples capas en conjunto con un algoritmo voraz (*greedy algorithm*) no da como resultado una máquina de Boltzmann[24] multicapa, en cambio, podría hablarse de una modelo generativo híbrido. Los autores denominan este modelo *red de creencia profunda* (*deep belief network*), puesto que las conexiones entre las unidades ocultas se encuentran restringidas de forma tal que dichas unidades forman múltiples capas. Esta configuración permite usar un grupo de máquinas de Boltzmann modificadas para inicializar los pesos de una máquina de Boltzmann profunda antes de iniciar el proceso de aprendizaje.

Bengio (2009) describe los enfoques antes mencionados como una metodología de entrenamiento de los niveles intermedios de representación (capas ocultas) mediante aprendizaje no supervisado que puede ser realizado de forma local en cada capa. Además, describe las arquitecturas profundas como aquellas compuestas por múltiples niveles de operaciones no lineales, tales como las redes neuronales con un grupo de capas ocultas o en implementaciones de fórmulas proposicionales complejas usando una serie de subformulas.

Bengio (2012) define *aprendizaje profundo* como la capacidad de aprender múltiples niveles de representación con la finalidad de descubrir atributos abstractos en los niveles superiores de la representación (red), de forma que sea más fácil delimitar y diferenciar cada atributo del resto de los elementos contenidos en los datos. A esta definición podemos sumar la propuesta por Schmidhuber (2015), quien define aprendizaje profundo como el resultado en conjunto de múltiples cadenas de etapas computacionales, donde cada etapa transforma la suma de los pesos de cada capa de una red para optimizar el proceso de aprendizaje.

De forma más reciente, LeCun et al. (2015) y Mnih et al. (2015) definen el aprendizaje profundo en función de su capacidad de representación y abstracción, es decir la capacidad que tienen de simplificar información compleja en datos más sencillos que, gracias al uso de múltiples niveles de procesamiento, pueden aprender datos complejos. Así, para LeCun et al. (2015) los métodos de aprendizaje profundo se obtienen mediante la composición de módulos no lineales que transforman las representaciones de un nivel determinado en representaciones de mayor abstracción. El conjunto de múltiples abstracciones de este tipo permite que se puedan aprender funciones complejas. Por otra parte, Mnih et al. (2015) consideran que las redes neuronales profundas están conformadas por múltiples capas que son usadas para construir representaciones que aumentan la abstracción de los datos. Este tipo de redes han hecho posible que las redes neuronales sean usadas para aprender conceptos tales como la clasificación de objetos a partir de datos sensoriales.

Desde el desarrollo del MLP entrenado con retroprogación de errores al día de hoy, podemos destacar los siguientes cambios en materia de aprendizaje automático: se dispone de una mayor cantidad de datos y capacidad de procesamiento y, por lo tanto, es posible entrenar modelos neuronales con un mayor número de dimensiones. En nuestro caso, ve-

[24] *Restricted Boltzmann machine* en inglés o RBM por sus siglas.

remos cómo se aplica esta metodología en el campo del PLN, específicamente para llevar a cabo el entrenamiento de modelos neuronales que nos sirvan como representaciones del conocimiento contenido en la lengua general.

Las representaciones vectoriales de palabras tienen la finalidad de sustituir a los diccionarios como instrumentos de representación del vocabulario de la lengua general ya que, tal como se ha mostrado en los acercamientos previos y en la sección 2.2, la capacidad de representar computacionalmente una palabra del vocabulario de una lengua depende en gran medida de la calidad de las definiciones de los diccionarios de referencia. Mediante el uso de modelos neuronales de lengua se espera cubrir el vacío de información que algunas palabras pueden tener mediante la creación de grupos de similitud de cada una de las palabras del vocabulario de la lengua general.

5.7.1 Representaciones distribuidas de palabras y modelos neuronales de lengua

Antes de que Mikolov et al. (2013a) presentaran el popular modelo Word2Vec con las arquitecturas CBOW y Skip-Gram, Bengio et al. (2003), propusieron una metodología para crear modelos de lengua mediante redes neuronales. Esta técnica fue llamada *representaciones distribuidas de palabras* (*distributed representation for words* o *word embedding*[25] en inglés) y consiste en generar representaciones densas de palabras en vectores de dimensiones pequeñas y fijas.

Mediante estas representaciones distribuidas, un modelo de lengua tiene la capacidad de aprender un número exponencial de enunciados semánticamente similares. Este modelo puede aprender simultáneamente la representación distribuida para cada palabra de un vocabulario y la expresión de la función de probabilidad de cada secuencia de palabras expresada en términos de esta representación (Bengio et al., 2003, p. 1137).

Este modelo de representaciones distribuidas toma como entrada $w_1, \ldots, w_t$ palabras de $w_t \in V$, donde V es un vocabulario extenso, pero finito de tamaño $|V|$. El objetivo es aprender una función $f(w_t, \ldots, w_{t-n+1}) = \hat{P}(w_t|w_1^{t-1})$ tal que el modelo de como resultado altas probabilidades fuera de la muestra. Esta arquitectura maximiza la ecuación 5.33, este diseño posteriormente sería conocido como un modelo de lenguaje neuronal prototípico entrenado mediante entropía cruzada. El entrenamiento se lleva a cabo buscando un valor θ que maximice el corpus de entrenamiento penalizado con verosimilitud logarítmica donde $R(\theta)$ es un término de regularización.

$$L = \frac{1}{T} \sum_t \log\ f(w_1, w_{t-1}, \ldots, w_{t-n+1}; \theta) + R(\theta) \tag{5.33}$$

La función f (ver figura 5.21) es una composición de las representaciones de C y g, donde C se comparte en todas las palabras del contexto. Cada una de estas dos partes se encuentran asociadas a los parámetros de la representación C, donde cada parámetro es uno de atributos representados por una matriz $|V| \times m$, cuya fila i es el vector de atributos $C(i)$ de la palabra i. La función g puede ser implementada en paralelo mediante una red neuronal recurrente (RNN) alimentada hacia adelante con parámetros ω. De forma que el parámetro general se establece en $\theta = (C, \omega)$.

[25] En la práctica ambos términos suelen usarse de forma indistinta.

Los resultados de esta implementación fueron la obtención de un 24 % en perplejidad con el corpus Brown (un millón de palabras) y 8 % con el corpus AP News (15 millones de palabras), en comparación con modelos basados n-gramas. Los autores detallan que este incremento puede deberse a que, gracias a las representaciones distribuidas de palabras, se resuelve el problema de la dimensionalidad, ya que cada enunciado entrenado aporta información de otras combinaciones de enunciados al modelo.

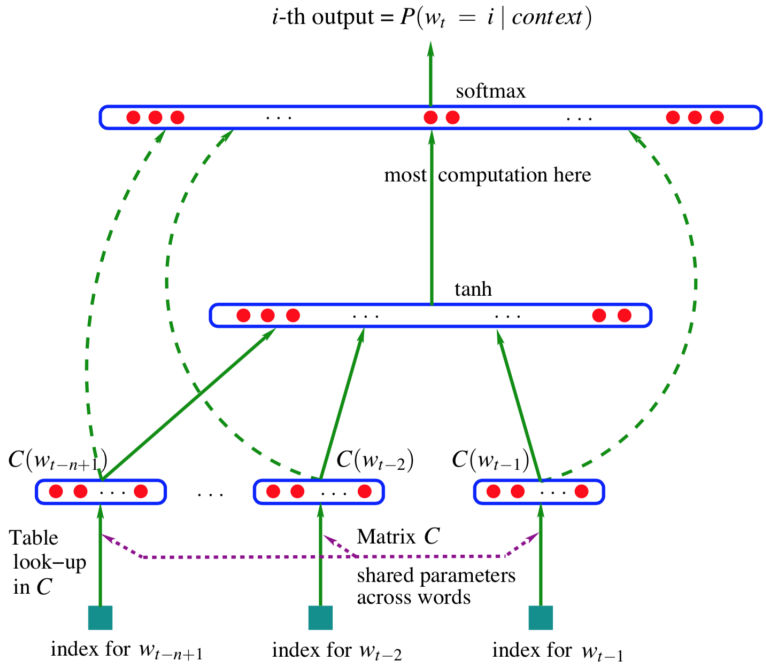


Figura 5.21 – Arquitectura neuronal $f(i, w_{t-1}, \ldots, w_{t-n+1}) = g(i, C(w_{t-1}), \ldots, C(w_{t-n+1}))$ donde g es la red neuronal y $C(i)$ es el vector de atributos de palabra con índice i (Bengio et al., 2003, p. 1142).

Por otra parte, Collobert y Weston (2008) introducen la idea de modelos de lengua previamente entrenados mediante aprendizaje profundo para resolver tareas comunes de PLN: *semantic role labeling* (SRL), *named entity recognition* (NER), etiquetado gramatical, *chunking* y generación de modelos de lengua. El modelo de lengua fue la primera parte de este modelo en ser entrenada ya que, según se describe en la figura 5.22, la metodología de entrenamiento implementa una capa con una tabla de búsqueda.

La primera capa de esta red extrae los atributos de cada palabra, la segunda capa extrae los atributos de los enunciados tratándolos como una secuencia con estructura global y estructura local, es decir, no se trata como bolsa de palabras (en inglés *bag of words* o BOW por sus siglas) y, finalmente, las últimas son capas neuronales convencionales (Collobert y Weston, 2008, p. 161). En este modelo (ver ecuación 5.34) una tabla de búsqueda LT_w tiene una representación vectorial para cada componente i que se encuentra dentro del vocabulario $\mathcal{D}$. Bajo este supuesto entendemos que $W \in \mathbb{R}^{d \times |\mathcal{D}|}$ es, entonces, una matriz de parámetros que debe ser aprendida y, por consiguiente, $W \in \mathbb{R}^d$ es la columna con índice i de W y d es la representación vectorial.

$$LT_w(i) = W_i \tag{5.34}$$

Los autores encontraron que la implementación de una segunda capa previamente entrenada, implementada como una tabla de búsqueda, reduce tiempo de procesamiento y simplifica el entrenamiento de los modelos de lengua. Esta nueva arquitectura de entrenamiento, en conjunto con las representaciones distribuidas de palabras, sentaron las bases para el desarrollo de modelos como Word2Vec (Mikolov et al., 2013c,a,b; Le y Mikolov, 2014), GloVe (Pennington et al., 2014), y FastText (Joulin et al., 2016a,b; Bojanowski et al., 2016).

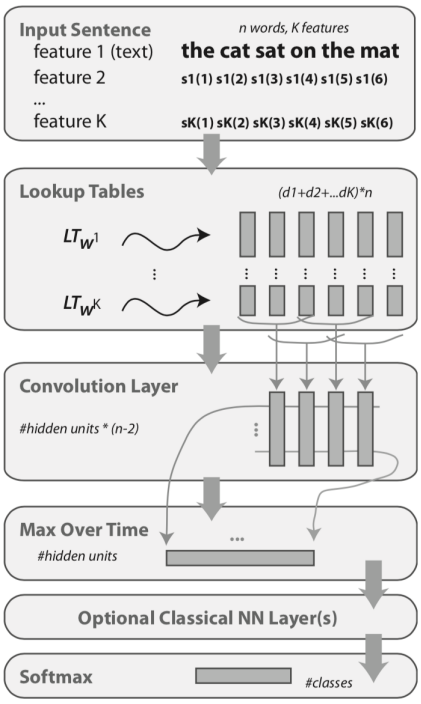


Figura 5.22 – Descripción de la arquitectura profunda neuronal (Collobert y Weston, 2008, p. 162).

En la actualidad las propuestas de Bengio et al. (2003) y Collobert y Weston (2008) han evolucionado a la par de los avances en materia de aprendizaje automático, por ejemplo: entrenar modelos mediante el uso de redes LSTM[26] (Jozefowicz et al., 2016) en la segunda capa de entrenamiento y la implementación de estos modelos como la base de los sistema de PLN modernos (Lebret, 2016), además de implementaciones de redes neuronales recurrentes muy profundas (Conneau et al., 2017).

5.7.2 Modelo Word2Vec

Tal como mencionamos en la sección 5.7.1, las representaciones distribuidas de palabras son un mecanismo que nos permite obtener información más detallada (densa) sobre las palabras en el contexto de un texto en comparación con modelos anteriores que simplemente agrupaban las palabras de un texto sin considerar su contexto de aparición.

[26]Redes de gran memoria de corto plazo, *long-short term memory* en inglés.

La primera propuesta de Mikolov et al. (2013c) en materia de *word embedding*, consistía en el uso de una RNN (ver figura 5.23) que simplificaba el proceso de entrenamiento y generalización propuesto por Bengio et al. (2003) y Collobert y Weston (2008). En esta arquitectura de RNN, el vector de entrada $\mathbf{w}(t)$ representa cada palabra de entrada en un tiempo t codificada empleando *1-of-N*[27], mientras que la capa de salida $\mathbf{y}(t)$ produce la distribución de probabilidad de las palabras. Finalmente, la capa oculta funciona como una capa de almacenamiento que mantiene las representaciones $\mathbf{s}(t)$. El vocabulario es igual al tamaño los vectores $\mathbf{w}(t)$ de la capa de entrada e $\mathbf{y}(t)$ de la capa de salida.

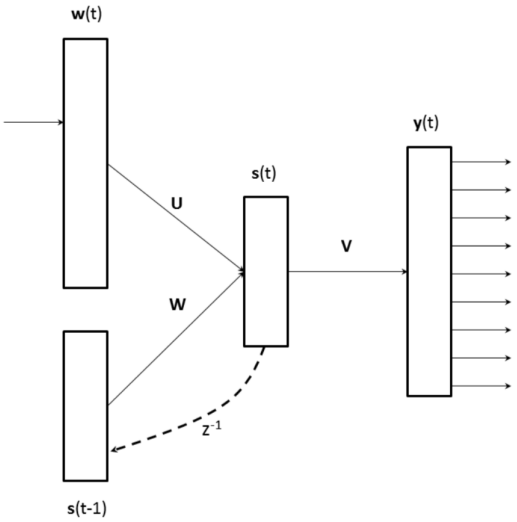


Figura 5.23 – Modelo de lengua mediante RNN (Mikolov et al., 2013c).

El modelo fue entrenado con retropropagación (*backpropagation*) para maximizar la probabilidad logarítmica de los datos y así, como resultado de este proceso, en la capa de salida se obtienen las representaciones (o *embeddings*) de cada palabra del vocabulario. La capa oculta $s(t)$ y la capa de salida $\mathbf{y}(t)$ se calculan siguiendo las Ecuaciones 5.35 y 5.36:

$$\mathbf{s}(t) = f(\mathbf{U}\mathbf{w}(t) + \mathbf{W}\mathbf{s}(t-1)) \tag{5.35}$$

$$\mathbf{y}(t) = g(\mathbf{V}\mathbf{s}(t)), \tag{5.36}$$

donde $\mathbf{U}$ contiene la representación generada de cada palabra del vocabulario del modelo y $f(z)$ y $g(z_m)$ se definen de acuerdo a 5.37.

$$f(z) = \frac{1}{1+e^{-z}}, \quad g(z_m) = \frac{e^{Z_m}}{\sum_k e^{z_k}}. \tag{5.37}$$

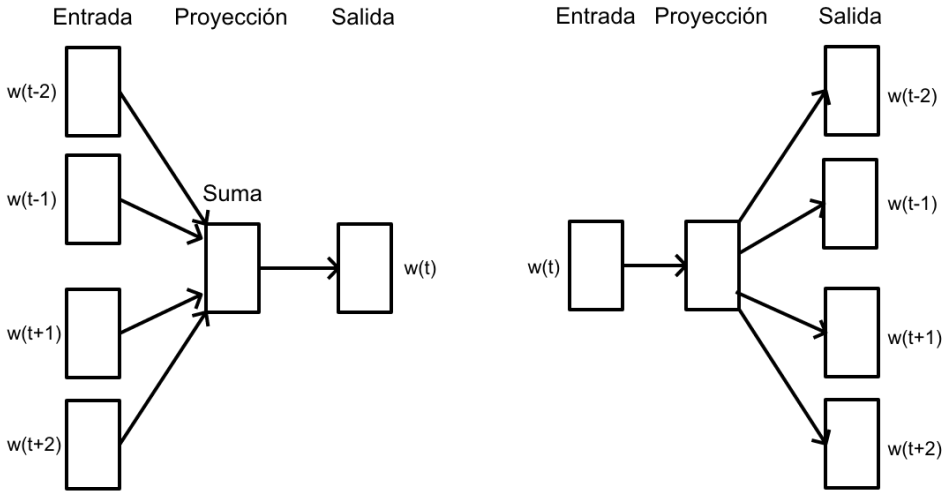
[27]En este tipo de representaciones se crean vectores con valores 0 y 1. Se asigna 1 a la palabra que está siendo representada y 0 al resto de los valores del vector. La extensión del vector es igual a N que es la extensión del vocabulario del modelo. Este tipo de representación también suele llamarse *one-hot-encoding*.

Para realizar la evaluación de este modelo, Mikolov et al. (2013a) propusieron dos tipos de evaluaciones: sintácticas y semánticas. La evaluación sintáctica consistió en comprobar la capacidad que tiene el modelo para completar analogías del tipo "¿A es a B como C es aX?", mientras que la semántica consistió en completar analogías por similitud del tipo "clase inclusiva: objeto singular", por ejemplo, "vestimenta: camisa". Los autores obtuvieron resultados superiores al estándar de aquel entonces, 40 % de casos correctos en el desempeño sintáctico, 0.275 en p de Spearman y precisión MaxDiff (Marley y Louviere, 2005; Louviere et al., 2015) con un valor de 0.418.

Word2Vec (Mikolov et al., 2013a) es un modelo que se basa en la primera propuesta de RNN de Mikolov et al. (2013c). Este nuevo modelo presenta mejoras tales como: reducción del tiempo de entrenamiento, incremento de conocimiento adquirido, dos nuevas arquitecturas de entrenamiento, además de simplificación del modelo de aprendizaje:

> The main observation [...] was that most of the complexity is caused by the non-linear hidden layer in the model. While this is what makes neural networks so attractive, we decided to explore simpler models that might not be able to represent the data as precisely as neural networks, but can possibly be trained on much more data efficiently (Mikolov et al., 2013a, p. 4).

La primera de las dos arquitecturas fue nombrada *Continous bag of words* (CBOW), esta arquitectura es similar al modelo neuronal alimentado hacia adelante (ver figura 5.24 *a*), donde la capa no lineal es eliminada y la capa de proyección es compartida por todas las palabras del vocabulario. El objetivo de aprendizaje en la arquitectura CBOW (ecuación 5.41) es la predicción de una palabra en función de una serie de atributos o palabras de entrada, sin que el orden de estas palabras influya en las proyecciones generadas.



(a) CBOW (b) Skip-Gram

Figura 5.24 – Tipos de entrenamiento con Word2Vec.

La arquitectura CBOW cuenta con un vocabulario con una extensión V y una capa oculta de tamaño N. La entrada es un vector codificado con *one hot encoding* en el cual solo una de sus unidades V, $\{x_1, \ldots, w_v\}$ es igual a uno, mientras que el resto es igual a 0. Los pesos entre la capa de entrada y la de salida se expresan como una matriz $\mathbf{W}$ con

dimensiones $V \times N$, donde cada entrada de esta matriz es la representación vectorial con $n - dimensiones$ de una palabra $\mathbf{v}_w$ y por lo tanto la entrada i de $\mathbf{W}$ es igual a $\mathbf{v}_{w_I}^T$, tal como se muestra en la ecuación 5.38:

$$\mathbf{h} = \mathbf{W}^T\mathbf{x} = \mathbf{W}_{(k,\cdot)}^T := \mathbf{v}_{w_i}^T \tag{5.38}$$

Desde la capa oculta hacia la capa de salida hay una matriz de pesos $\mathbf{W}' = \{w'_{ij}\}$ y, mediante esos pesos, podemos calcular un índice u_j (ecuación 5.39) para cada palabra del vocabulario generado, donde $\mathbf{v'}_{wj}$ es la columna con índice j de la matriz $\mathbf{W}'$.

$$u_j = \mathbf{v'}_{w_j}^T\mathbf{h} \tag{5.39}$$

Sustituyendo estos valores en Softmax (ver ecuación 5.40),

$$\text{softmax}(z)_i = \frac{\exp(z_i)}{\sum_j \exp(z_i)} \tag{5.40}$$

obtenemos la arquitectura CBOW, tal como se muestra en la ecuación 5.41:

$$p(w_j|w_I) = \frac{\exp(\mathbf{v'}_{w_j} T_{\mathbf{v}_{w_I}})}{\sum_{j'=1}^{V} \exp(\mathbf{v'}_{w_{j'}} T_{\mathbf{v}_{w_I}})} \tag{5.41}$$

La segunda arquitectura, Skip-Gram (ver figura 5.24 *b*) tiene como objetivo encontrar representaciones de palabras que sean útiles para predecir las palabras que rodean a un enunciado. Siguiendo la notación propuesta para la arquitectura CBOW podemos definir la arquitectura Skip-Gram a partir de la ecuación 5.42:

$$\mathbf{h} = \mathbf{W}_{(k,\cdot)}^T = \mathbf{v}_{w_I}^T \tag{5.42}$$

En la capa de salida se calculan C distribuciones multinomiales, de forma que se obtiene la ecuación 5.43, donde $w_{c,j}$ es la palabra con índice j de la entrada x en la capa de salida. Por otra parte, $w_{O,c}$ es una palabra con índice c en el contexto de palabras de salida y w_I es la única palabra de entrada.

$$p(w_{cj} = w_{O,c}|w_I) = y_{c,j} = \frac{\exp(u_{c,j})}{\sum_{j'}^{V} \exp(u_{j'})} \tag{5.43}$$

La variable $y_{c,j}$ es la salida de la unidad j de cada elemento c de la capa de salida y $u_{c,j}$ es la entrada de cada elemento j en los elementos c en la capa de salida. Por lo tanto, como se puede observar en la ecuación 5.44, $\mathbf{v'}_{wj}$ representa el vector de salida de una palabra j del vocabulario.

$$u_{c,j} = u_j = \mathbf{v'}_{wj}^T \cdot \mathbf{h} \tag{5.44}$$

Sustituyendo 5.44 en 5.43 obtenemos la ecuación 5.45:

$$p(w_{O,c}|w_I) = \frac{\exp({\mathbf{v'}_{w_j}}^T \cdot \mathbf{h})}{\sum_{j'=1}^{V} \exp({\mathbf{v'}_{w_j}}^T \cdot \mathbf{h})} \tag{5.45}$$

Skip-Gram emplea la función de activación Softmax jerárquico (ver ecuación 5.46) (*hierarchical Softmax*) en conjunto con representaciones del vocabulario basadas en árboles binarios de Huffman (1952) para obtener la probabilidad $p(w_O|w_I)$ en la capa de salida. Los árboles de Huffman asignan valores pequeños a las palabras más frecuentes del vocabulario, para así reducir el tiempo de procesamiento gracias a que solo requiere $\log_2(\text{unigrama} - \text{perplejidad}(V))$ para realizar su evaluación. Esta función no cuenta con un vector de representación de palabras como salida sino que, cada unidad interna $V-1$ tiene un vector de salida $\mathbf{v'}_{n(w,j)}$.

$$p(w = w_O) = \prod_{j=1}^{L(w)-1} \sigma\left([\![n(w,j+1) = \text{ch}(n(w,j))]\!] \cdot \mathbf{v'}_{n(w,j)} \top \mathbf{h}\right) \tag{5.46}$$

Por lo tanto, definimos $\text{ch}(n)$ como el hijo de la unidad n, $\mathbf{v'}_{n(w,j)}$ es el vector de salida de la unidad $n(w,j)$ y $\mathbf{h}$ representa el valor de salida de la capa oculta definida como $\mathbf{h} = \mathbf{v}_{wI}$. La función de activación Softmax jerárquico convierte los valores de los pesos de cada neurona de la capa donde es aplicado en valores de probabilidad que suman 1. Por ejemplo, un modelo entrenado con 100 neuronas y un vocabulario de 10.000 palabras en la capa oculta, da como resultado en la capa de salida 10.000 representaciones (una por cada palabra del vocabulario) con las 100 probabilidades de las palabras que pueden ocurrir aleatoriamente con la palabra que está siendo representada.

Podemos observar en el ejemplo anterior que la capa oculta tendría una dimensión de 10,000 palabras por 100 neuronas que dan un total 1,000,000 de pesos. Esta red crece exponencialmente y, por lo tanto, si queremos obtener representaciones más densas (mayor cantidad de neuronas) o si tenemos un vocabulario más extenso, debemos considerar que el aumento de tamaño de está red también implica aumento en el tiempo de cálculo.

Para agilizar el tiempo de procesamiento, Mikolov et al. (2013b) proponen dos técnicas de entrenamiento nuevas llamadas submuestreo de palabras comunes y muestreo negativo simplificado (en inglés *negative sampling*). La primera técnica consiste en eliminar cada palabra w_i del corpus de entrenamiento mediante la ecuación 5.47, donde $f(w_i)$ es la frecuencia de cada palabra w_i. Mediante este proceso se descartan palabras que no son informativas durante el entrenamiento (artículos, preposiciones, etc.), pero se mantienen dentro del vocabulario y por lo tanto cuentan con representaciones.

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}} \tag{5.47}$$

La segunda técnica, el muestreo negativo (ecuación 5.48), permite agilizar el proceso de entrenamiento, específicamente durante el ajuste de los pesos en cada etapa de entrenamiento. Siguiendo con los valores presentados en el ejemplo anterior, los 1,000,000 de pesos se deberían reajustar durante cada ciclo de entrenamiento para cada muestra. El

negative sampling[28] consiste en seleccionar un número fijo de atributos k para asignarles valores negativos y de esta forma aumentar la probabilidad de la palabra objetivo wO del conjunto $P_n(w)$.

$$\log p(w|w_I) = \log \sigma(w'_{wO}{}^{\top} w_{WI} + \sum_{i=1}^{k} \mathbb{E}_{wi \sim P_n(w)}[\log \sigma(-v'_{wi}{}^{\top} v_{wI})]) \tag{5.48}$$

Los autores proponen un número k de entre 2 y 5 palabras en el caso de corpus extensos y un valor de k entre 5 y 20 para corpus pequeños (Mikolov et al., 2013b). Seleccionando $k = 5$ se tendrían que actualizar 600 pesos en la red de ejemplo, de estos 600 pesos que serán actualizados: 100 corresponden a la palabra que se desea representar y 500 pesos corresponden a los pesos de las palabras que se reducirán. Podemos observar que esta metodología reduce significativamente la cantidad de pesos que deben ser actualizados durante cada ciclo de entrenamiento, de 1,000,000 a 600.

Ambos modelos crean representaciones para cada palabra del vocabulario siendo la entrada el vector con la representación 1-de-N de una palabra nodo y la salida un vector con las probabilidades que tiene cada palabra de aparecer en el contexto de dicha palabra nodo. Los *embeddings* codifican relaciones semánticas que se construyen en el discurso entre las palabras del vocabulario (ver figura 5.25), es decir, las palabras que tienen mayor similitud entre sí tienen representaciones vectoriales similares ya que son definidas empleado palabras y términos similares.

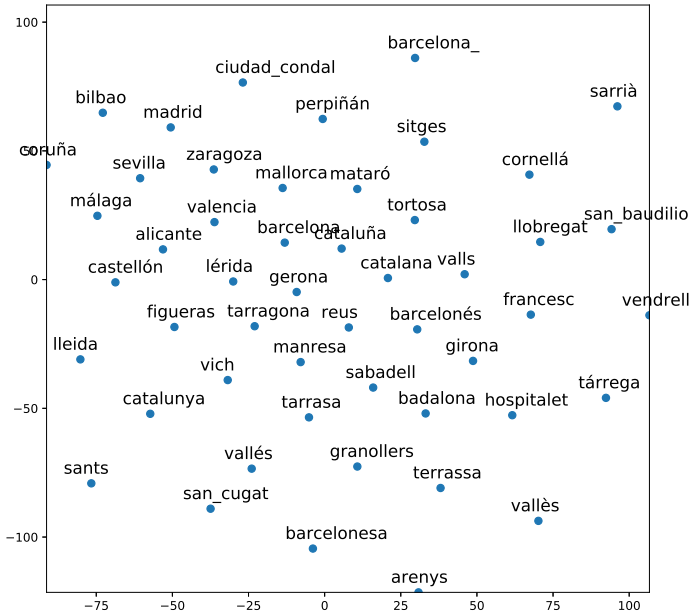


Figura 5.25 – Representación en componentes principales de las 50 palabras más similares a “Barcelona” generada con nuestro modelo Word2Vec en español.

[28]Cabe señalar que los autores no proporcionan una explicación profunda de los modelos matemáticos que presentan, sin embargo en Goldberg y Levy (2014) encontramos una derivación detallada del proceso de muestreo negativo.

Para realizar la desambiguación de significado que proponemos, estamos interesados en obtener *embeddings* que contengan palabras que representen los significados más comunes (semánticamente similares) de las palabras que componen el vocabulario de la lengua general. Para este fin entrenamos nuestros modelos Word2Vec empleando los corpus Wikipedia en cada lengua de trabajo (descritos en la sección 5.2.1), todos los corpus fueron procesados siguiendo los mismos pasos:

- Limpieza de etiquetas y caracteres especiales.
- Segmentación y tokenización.
- Creación de modelo de n-gramas.
- Entrenamiento de modelo Word2Vec.

Entrenamos nuestros modelos bajo la arquitectura Skip-Gram con 300[29] dimensiones durante 20 épocas o iteraciones empleando el modelo de trigramas generado en cada lengua. Como parámetros de configuración, usamos una ventana de contexto igual a 5 palabras a la izquierda de la palabra principal y 5 palabras a la derecha, la aparición mínima de cada palabra fue configurada a 20 ocurrencias y usamos muestreo negativo con un valor de $k = 5$.

Las representaciones obtenidas permiten observar las relaciones que existen entre las palabras del vocabulario de cada corpus. Podemos, por ejemplo, al analizar los agrupamientos de palabras antónimas como *salud* y *enfermedad* (figura 5.26), que muestran dos grupos de palabras separables que comparten algunos elementos en común. Las palabras que forman el agrupamiento de *salud* y el agrupamiento de *enfermedad* son las 150 palabras que, mediante el cálculo de la similitud por coseno, son más similares a *salud* y a *enfermedad*.

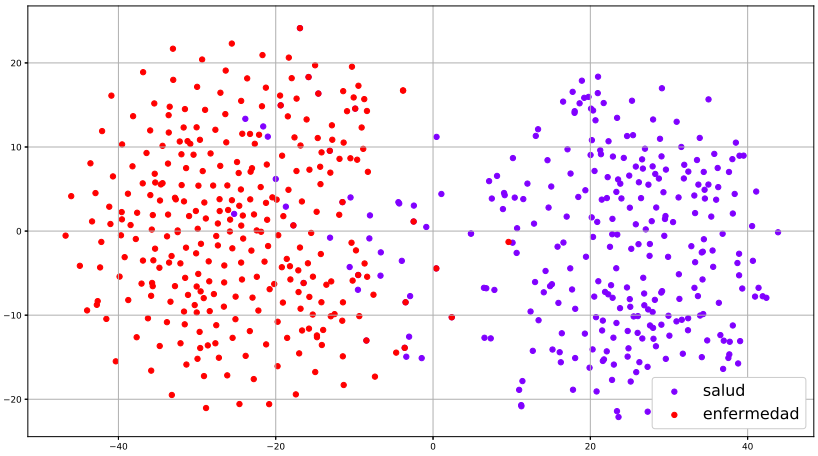


Figura 5.26 – Espacio vectorial de *salud* y *enfermedad*.

[29]Optamos por usar los parámetro implementados por Mikolov et al. (2013b), Pennington et al. (2014) y Bojanowski et al. (2016) quienes establecen 300 como un valor base. Yin y Shen (2018) considera que el uso de 300 dimensiones se debe a la influencia que han tenido las tres publicaciones antes mencionadas (principalmente Mikolov et al. (2013b)) y también propone una metodología para seleccionar la dimensionalidad de un modelo en función de la extensión del corpus de entrenamiento.

También podemos ver las representaciones de palabras sinónimas, como el caso de *browser* y *navegador* (figura 5.27) que comparten un espacio muy similar ya que ambas palabras suelen aparecer en los mismos contextos. Tanto este caso como el mencionado anteriormente, son ejemplos de las similitudes que se pueden obtener mediante las representaciones vectoriales de palabras.

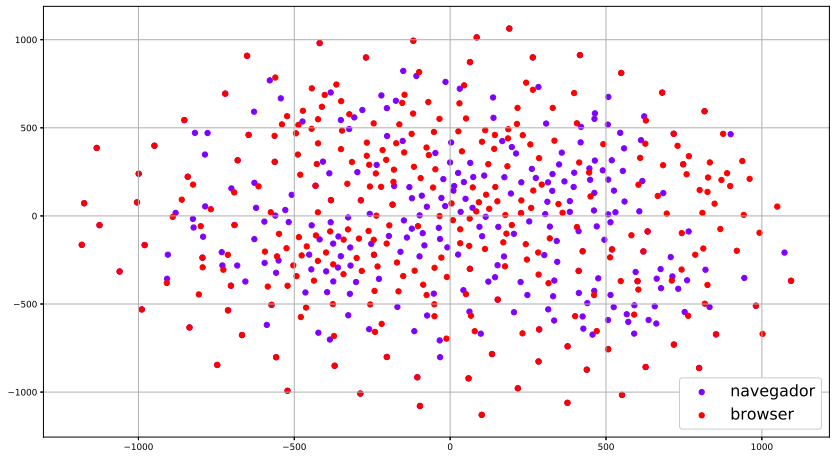


Figura 5.27 – Espacio vectorial de *navegador* y *browser*.

Los autores expanden el modelo Word2Vec en Le y Mikolov (2014), donde exploran la idea de generar representaciones de documentos y de párrafos mediante la aplicación de la arquitectura Skip-Gram a un documento entero o párrafo. Bajo esta arquitectura un documento es representado como una matriz que es acompañada de las palabras asociadas semánticamente a dicho documento. A pesar de las ventajas que señalan los autores (sobre todo en materia de análisis de sentimiento y clasificación de documentos), este modelo no se adapta a las necesidades del proyecto.

Word2Vec ha servido como punto de partida para el desarrollo de otros modelos neuronales de lengua, tal como es el caso de los modelos Sense2Vec (Trask et al., 2015) y FastText (Bojanowski et al., 2016). El primero es una implementación de Word2Vec con la ventaja añadida de usar texto etiquetado gramaticalmente, de forma que las representaciones generadas cuenten con una etiqueta gramatical asociada. El segundo, FastText, también construye sobre los fundamentos presentados por Mikolov et al. (2013b), pero busca generar representaciones empleando los componentes de cada palabra que será representada, generando así representaciones para elementos fuera del vocabulario.

5.7.3 Modelo FastText

FastText (Bojanowski et al., 2016) también emplea redes neuronales para generar representaciones de palabras con la particularidad de emplear información a nivel carácter de palabra como las unidades mínimas a ser entrenadas, a diferencia del modelo Word2Vec cuyas unidades mínimas de entrenamiento son palabras. En un modelo FastText el vector de la palabra *viral* estaría compuesto de los n-gramas internos de *viral*, acompañada de los caracteres especiales “<” y “>” como delimitadores, de la siguiente manera: “<v”, “vir”, “vira”, “viral”, “viral>”, “ira”, “iral>”, “iral>”, “ral”, “ral>”, “al>”.

Bojanowski et al. (2016) emplean como punto de partida la arquitectura Skip-Gram de Mikolov et al. (2013b) para llevar a cabo el proceso de aprendizaje a nivel de carácter. Generalmente, el proceso de entrenamiento en una arquitectura Skip-Gram produce como resultado vectores con representaciones de cada palabra del vocabulario del modelo. No obstante, mediante la adición de la ecuación 5.49, el modelo FastText es capaz de aprender representaciones de palabras empleando la información que aportan los n-gramas que componen cada palabra.

$$s(w, c) = \sum_{g \in G_w} \mathbf{z}_g^\top \mathbf{v}_c. \tag{5.49}$$

El modelo es entrenado bajo el supuesto siguiente: dado un diccionario de n-gramas con una dimensión G y dada una palabra w, describimos el conjunto de n-gramas que conforman w como el subconjunto $G_w \subset 1, \ldots, G$ y, finalmente, a cada n-grama g se asocia una representación vectorial $\mathbf{z}_g$ y vector de contexto $\mathbf{v}_c$. Mediante este supuesto, cada palabra es representada como la suma de las representaciones vectoriales de cada los n-gramas que la componen (Bojanowski et al., 2016). Gracias que FasText construye sobre la arquitectura Skip-Gram (sección 5.7.2) con muestreo negativo, pero en este caso, cada palabra es representada por sus componentes y, a su vez, por las palabras que aparecen en su contexto.

Los autores evaluaron esta implementación en nueve lenguas: árabe, checo, alemán, inglés, español, francés, italiano, rumano y ruso. Los autores evaluaron FastText mediante dos experimentos: la evaluación de analogías y la evaluación de juicio humano. FastText superó a otras metodologías en la evaluación de juicio humano de similitud. Los resultados de la evaluación de analogías se encuentran dentro del rango de valores de las arquitecturas CBOW y Skip-Gram (ver tabla 5.11), sin obtener porcentajes superiores en ningún caso. La evaluación de analogías es de particular interés para esta tesis, puesto que se interpreta como un indicador de la exactitud sintáctica y semántica del modelo. Habría que valorar la pertinencia de implementar un modelo más complejo (FastText) sobre uno más simple (Word2Vec).

		Skip-Gram	CBOW	FastText
Checo	Semántico	25.7	27.6	27.5
Alemán	Semántico	66.5	66.8	62.5
Inglés	Semántico	78.5	78.2	77.8
Italiano	Semántico	52.3	54.7	52.3

Tabla 5.11 – Fragmento de la tabla de resultados en la evaluación de analogía de palabras de Bojanowski et al. (2016, p. 5).

En el caso de este modelo no realizamos un entrenamiento propio, sino que empleamos los modelos previamente entrenados bajo la arquitectura CBOW[30] que se encuentran disponibles en la página de FastText[31]. Estos modelos fueron entrenados con la arquitectu-

[30]Estos modelos no son los descritos en Joulin et al. (2016a), sino que son una versión actualizada empleando la arquitectura CBOW.

[31]`https://FastText.cc/docs/en/crawl-vectors.html`

ra CBOW con pesos de posición, empleando 300 dimensiones, con n-gramas de caracteres de una longitud igual a 5, en una ventana con tamaño de 5 positivos y 10 negativos.

5.7.4 Modelo Sense2Vec

En las secciones anteriores hemos abordado los modelos de representaciones de palabras Word2Vec y FastText. Ambos modelos emplean diferentes estrategias para llegar a un fin común: generar presentaciones densas de palabras en un espacio vectorial. Sin embargo, hasta ahora hemos hablado de una representación (o un vector) por elemento del vocabulario, esto se debe a que ambos modelos no cuentan con una estrategia para agrupar o detectar los diferentes significados que puede tener una palabra.

El modelo Sense2Vec que proponen Trask et al. (2015) intenta solventar esta limitación mediante el uso de un corpus etiquetado gramaticalmente como datos de entrenamiento, para así obtener un nivel adicional de representación: un vector por cada etiqueta gramatical asignada a cada palabra del vocabulario, generando así más de una representación por palabra.

En la figura 5.28 podemos observar que la estructura del modelo Sense2Vec es muy similar al modelo de Mikolov et al. (2013a), con la particularidad de que Trask et al. (2015) buscan agrupar los diferentes significados que puede tener cada palabra del vocabulario en función de su etiqueta gramatical. Esta metodología se inspiró en el trabajo de Huang et al. (2012), cuya metodología emplea estrategias de aprendizaje automático no supervisado para generar múltiples representaciones para cada palabra, en concreto, una representación por cada agrupamiento generado.

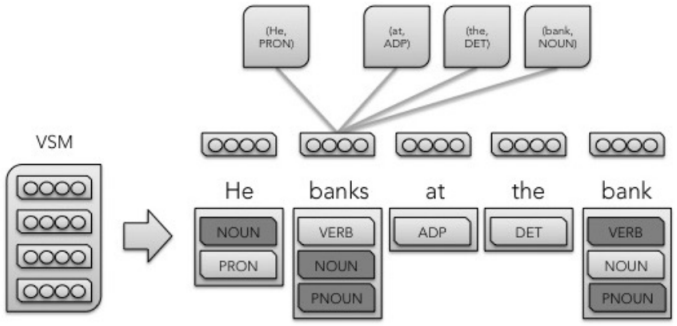


Figura 5.28 – Descripción del modelo Sense2Vec, Trask et al. (2015).

El modelo Sense2Vec se construye sobre los principios metodológicos de Word2Vec. No obstante, este tiene dos diferencias principales, la primera radica en los datos de entrada: antes de iniciar el proceso a entrenamiento de la red se debe etiquetar gramaticalmente el corpus. La segunda consiste en que, una vez obtenido el corpus etiquetado, el modelo Sense2Vec se puede entrenar mediante cualquiera de las dos arquitecturas de Word2Vec (Skip-Gram o CBOW), pero durante el proceso de entrenamiento el algoritmo realiza las predicciones de las etiquetas gramaticales en función de las etiquetas gramaticales del contexto de cada palabra.

Para obtener un modelo empleamos como datos de entrada los corpus de Wikipedia en catalán, español y francés que fueron utilizados para entrenar el modelo Word2Vec,

ya que ambos modelos comparten la misma arquitectura. En este caso realizamos el entrenamiento con un modelo de unigramas etiquetados, dado que al realizar el etiquetado del modelo de n-gramas hubo una pérdida de precisión. Para llevar a cabo el etiquetado gramatical, usamos los modelos de lengua en español y francés de la librería spaCy y en el caso del catalán el modelo de lengua creado específicamente para esta tesis.

Como parámetros de entrenamientos nos apegamos a los descritos en la bibliografía: empleamos el juego de etiquetas de Universal Dependencies[32] durante el etiquetado gramatical y realizamos el proceso de edición de etiquetas siguiendo la estructura *palabra|etiqueta* para reconstituir las oraciones del corpus. Trask et al. (2015) utilizaron los siguientes parámetros de entrenamiento: 500 dimensiones, un tamaño de ventana de contexto igual a 5 palabras a ambos lados del nodo y un conteo mínimo de 10 ocurrencias por palabra bajo la arquitectura Skip-Gram. Sin embargo, hemos optado por implementar los mismos parámetros de entrenamiento que seguimos en el modelo Word2Vec, para mantener homogeneidad entre modelos y poder realizar paralelismos durante la evaluación de los modelos.

Los tres modelos de cada lengua son evaluados para seleccionar el modelo que genere las representaciones más adecuadas a nuestras necesidades: obtener un modelo neuronal que sirva como referencia de los significados de lengua general de cada palabra de nuestro vocabulario. Estos vectores servirán como sustituto de los diccionarios de referencia ya que —a pesar que los diccionarios documentan los posibles significados de una palabra— la estructura sintáctica de una definición no siempre aporta información suficiente para desambiguar automáticamente el significado de una palabra.

5.8 Elementos seleccionados para el desarrollo de sistema DENISE

Tras llevar a cabo las evaluaciones previas de los diferentes métodos, modelos de clasificación de documentos y algoritmos de extracción de palabras claves, se seleccionaron los siguientes elementos para ser evaluados en profundidad:

- Detección de tema:
  - Regresión logística.
  - Perceptrón multicapa.
  - Clasificador SVM.
- Extracción de palabras claves: algoritmo TextRank con filtros gramaticales que puede implementarse en las tres lenguas de trabajo y las lenguas disponibles en las bibliotecas spaCy de Python.
- Representaciones vectoriales de palabras para la desambiguación de significado y tema:
  - Modelo Word2Vec: representaciones de palabras basadas en la arquitectura Skip-Gram.

[32] `http://universaldependencies.org/u/pos/`

- Modelo FastText: representaciones de palabras basadas en la arquitectura CBOW.
- Modelo Sense2Vec: representaciones de palabras con etiquetas gramaticales basadas en la arquitectura Skip-Gram.

La intención de evaluar tres modelos de representaciones de palabras es comprobar la eficacia que tiene cada modelo. El modelo Word2Vec puede ser considerado un enfoque convencional, ya que en la actualidad se usa con frecuencia en diversas aplicaciones de procesamiento del lenguaje natural. El modelo FastText añade un nivel de procesamiento adicional, genera vectores tomando en cuenta los componentes de cada elemento del vocabulario. Y, finalmente, el modelo Sense2Vec emplea etiquetas gramaticales para generar representaciones adicionales para aquellas palabras que son usadas con diferentes funciones gramaticales.

Capítulo 6

EVALUACIONES Y ANÁLISIS

El proceso de evaluación del sistema se divide en dos etapas, una etapa individual donde se evalúa la precisión y exhaustividad de cada modelo de clasificación; y una segunda etapa de evaluación del flujo de trabajo completo del sistema con datos nuevos. Para llevar a cabo la primera etapa de evaluación, emplearemos dos subconjuntos (uno por cada lengua de trabajo) de concordancias obtenidas del OBNEO con NS provenientes de la informática.

En total se obtuvieron 194 concordancias de un total de 5,562 en español y 120 concordancias de un total de 3,709 en catalán. En las figuras 6.1 y 6.2 se puede ver el formato de los datos de entrada, estos consisten en una tabla en formato CSV donde cada NS está separado por un tabulador y acompañado de su concordancia. Limitamos la evaluación de los métodos de clasificación y extracción de palabras a catalán y español puesto que no contamos un conjunto de datos paralelo o equivalente en francés. La traducción automática de estas concordancias no resultó en un conjunto de datos viable, puesto que fue posible corroborar que las unidades que son neológicas en catalán y español, también lo sean en francés. Basándonos en este factor de incertidumbre, optamos por descartar el uso de un conjunto de datos en francés generado con traducción automática.

```
1 term	context
2 adobe photoshop	Inclús, bueno, hi han pàgines en blanc perquè si el client ho vol fer des del seu adobe photoshop pot incloure la
  seva pròpia maqueta dintre de lo que és el fotollibre.
3 antivíric -a	Cal prevenir les intromissions a través de la instal·lació de components hardware i software adients i cal
  prevenir l'actuació dels anomenats virus informàtics a través de la instal·lació de sistemes antivírics eficaços i fàcils de
  mantenir: tot un repte.
4 antivirus	La lectura dels historials clínics del virus i cavalls de Troia que duen tots els paquets antivirus és apassionant.
5 aplicació	Ahir vau comprar una aplicació del mòbil que fa meravelles però avui ni tan sols recordeu haver-li autoritzat l'accés
  a la llibreta de contactes.
6 aplicació	EF és l'única organització que desenvolupa el seu propi pla d'estudis, mètodes d'aprenentatge, llibres, cursos i
  aplicacions online.
7 aplicació	Es tracta d'una aplicació per silenciar automàticament els dispositius mòbils als llocs on es requereix silenci, és a
  dir, hospitals, sales d'espectacles, biblioteques, platós de ràdio, per exemple.
8 baixar	"""Val més que ho vegeu perquè això no és en cap disc ni es pot baixar d'internet""."
```

Figura 6.1 – Fragmento del conjunto de datos de prueba en catalán.

Empleando los NS que fueron detectados en cada concordancia, generamos un listado de NS que será usado durante las etapas de evaluación de los procesos de extracción de KW y desambiguación de significado. Estos NS fueron normalizados y lematizados, para comparar si existe una variación de resultados entre lemas y formas.

El presente capítulo se presenta de acuerdo al flujo de proceso del sistema. Comenzamos analizando la detección automática de lengua empleado Langdetect, en catalán y en español. Posteriormente evaluamos los modelos de clasificación automática seleccionados en las tres lenguas de trabajo para justificar la selección de un modelo de regre-

sión logística. Después, procedemos a evaluar nuestra implementación de TextRank para obtener precandidatos a neologismo semántico empleando los conjuntos de datos antes mencionados.

```
term	context
acelerar	quate, startup acelerada por Wayra Perú, la aceleradora de Telefónica Open Future, informó que está brindando servicios a más de 600 artistas en Latinoamérica, entre músicos y youtubers, y distribuye sus contenidos en más de 50 tiendas digitales y plataformas de streaming.
agujero	Tras su arresto, Mitnick fue trasladado a la cárcel del condado de Smithfield, en Carolina del Norte, donde pasó una semana en el agujero, es decir, en prisión incomunicada.
alfombrilla	Alfombrilla para ratón.
almacenado	Esta transmisión de conocimiento dentro de la empresa se logró con un sistema de almacenado de conocimiento implementado mediante una base de datos.
alojamiento	Visitar este portal es adentrarse en una comunidad de Internet desenfadada y gratuita, destinada al alojamiento de blogs: un formato de publicación en la Red tipo bitácora o diario personal, que está causando furor.
androide	HTC, LG, Samsung, Sony, Huawey o ZTE  se han convertido en los principales vendedores de androides los smartphones con sistema operativo de Android.
```

Figura 6.2 – Fragmento del conjunto de datos de prueba en español.

En la siguiente sección comparamos tres modelos de representaciones vectoriales de palabras en catalán y español mediante un problema de clasificación. Utilizamos los listados de neologismos semánticos previamente reconocidos para generar nuevos listados con las palabras más similares (campos semánticos) a cada neologismo. Tratamos cada listado como un documento al que se debe asignar una temática y, para este fin, revisamos manualmente las temáticas asignadas a cada documento, así como los campos semánticos de cada neologismo.

Finalmente, evaluamos la desambiguación de tema empleando nuestra metodología. Nuevamente empleamos el conjunto de datos candidato - concordancia generado a partir de la base de datos del OBNEO. Durante esta etapa mostramos un escenario experimental del funcionamiento de DENISE, para observar los posibles resultados de la implementación de nuestro sistema dentro de un entorno de trabajo.

6.1 Detección automática de lengua

Para evaluar la implementación de Langdetect introdujimos las concordancias de ambas lenguas al módulo, esperando que, a la totalidad de los contextos, se asigne la etiqueta de lengua que le corresponde. Un resultado correcto asigna un valor *Verdadero*, mientras que un resultado incorrecto asigna un valor *Falso*, para evaluar este proceso empleamos las siguientes métricas: precisión, exhaustividad, *f1-score* y soporte.

	Precisión	Exhaustividad	f1-Score	Soporte
Catalán	1.0	0.98	0.99	120
Español	1.0	0.97	0.99	194

Tabla 6.1 – Precisión, exhaustividad, *f1-score* y soporte por lengua de trabajo detectada.

El valor obtenido en precisión (ver tabla 6.1) nos indica que la totalidad de los casos fueron evaluados. Por otra parte, la exhaustividad muestra que se obtuvo un total de 98 % de verdaderos positivos en catalán y 97 % de verdaderos positivos en español. En ambos casos los resultados de *f1-score* fueron iguales a un 99 %. Estas cifras confirman que el total de los casos analizados cuentan con un alto índice de verdaderos positivos evaluados correctamente. Así, estos valores indican que este módulo opcional aporta un

método adecuado para agilizar el trabajo con múltiples fuentes, sin necesidad de asignar una etiqueta de lengua a cada texto.

6.2 Clasificación y detección de temas

En la sección 5.5 se llevó a cabo una evaluación preliminar de diversos modelos de clasificación y, como resultado de dicha evaluación, se seleccionaron los modelos siguientes como candidatos viables para la implementación final del sistema: regresión logística (LR), clasificador de máquina de vectores de soporte (SVC) y el perceptrón multicapa (MLP). Estos tres modelos obtuvieron valores de exactitud similares, no obstante, cada acercamiento tiene ventajas y desventajas, por ejemplo: el tiempo de entrenamiento, el tiempo de análisis y la capacidad de generalización.

Como metodología de evaluación de estos tres modelos calculamos el promedio de precisión usando una separación del corpus de entrenamiento en dos subconjuntos: 33 % del total del corpus corresponde al subconjunto de prueba y 66 % al subconjunto de entrenamiento. Esta métrica permitirá comparar la precisión obtenida con el subconjunto de entrenamiento de cada modelo, así como el tiempo total de procesamiento. También nos permitirá comparar los resultados del subconjunto de prueba que, al mismo tiempo, sirven como indicador de la capacidad de generalización de cada modelo.

Empleamos matrices de confusión para ilustrar los resultados obtenidos tras el entrenamiento de cada modelo de clasificación. Una matriz de confusión C es tal que $C_{i,j}$ es igual al número de observaciones conocidas que pertenecen al grupo i, pero que han sido clasificadas como pertenecientes del grupo j. Por lo tanto, en el caso de un clasificador binario, el total de los verdaderos negativos se encuentra representado por $C_{0,0}$, los falsos negativos $C_{1,0}$, verdaderos positivos $C_{1,1}$ y falsos positivos $C_{0,1}$.

En términos simples, los resultados de la diagonal de una matriz de confusión representarían los verdaderos positivos de cada clase de un modelo M. En el resto de las celdas de esta matriz encontramos los elementos clasificados incorrectamente. Es de esperar que existan errores en cada modelo ya que un valor igual al 100 % de precisión indicaría que el modelo entrenado ha sido sobreajustado. Un modelo sobreajustado suele tener como desventaja poca capacidad de generalización y, por lo tanto, buscamos valores de precisión que se encuentren por encima del 90 % sin llegar al 100 %.

6.2.1 Resultados de modelos de clasificación en catalán

El primer modelo a evaluar fue el SVC, dicho modelo obtuvo un total de 97 % de precisión promedio con el conjunto de entrenamiento y 95 % de precisión con el conjunto de prueba. En la figura 6.3 podemos observar que *informática* tiene seis casos mal clasificados, tres documentos fueron detectados como pertenecientes a *derecho* y dos documento como perteneciente a *economía*.

Por otra parte, el modelo MLP obtuvo 98 % de precisión promedio con el conjunto de prueba y 94 % con el conjunto de entrenamiento. Este modelo clasificó de forma incorrecta siete documentos (ver 6.4), tres fueron clasificados como perteneciente a *derecho*, uno como *medicina*, dos como *economía* y uno como *lingüística*.

Finalmente, el modelo LR obtuvo 96 % de precisión con el conjunto de entrenamiento y 93 % de precisión con el conjunto de prueba. A pesar de haber obtenido un porcentaje

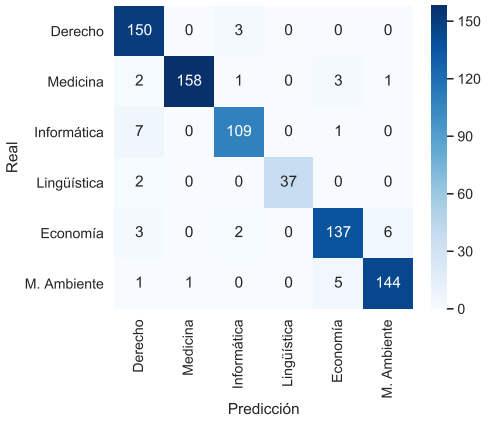


Figura 6.3 – Evaluación de predicciones de tema del modelo SVC en catalán.

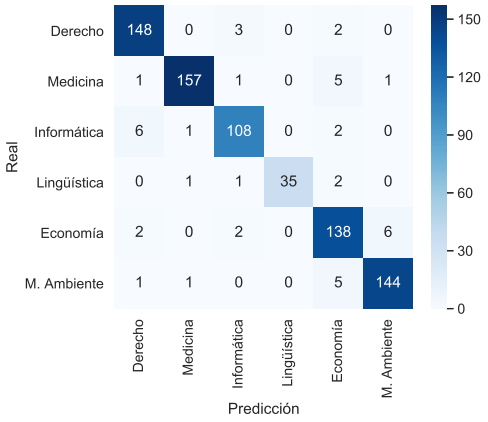


Figura 6.4 – Evaluación de predicciones de tema del modelo MLP en catalán

menor en comparación con los modelos MLP y SVC, el modelo LR clasificó incorrectamente (ver 6.5) siete documentos: cuatro documentos en *derecho*, uno en *medicina*, dos como *lingüística* y tres documentos como *economía*.

Como podemos ver en la tabla 6.2, los resultados son bastante similares, hay una diferencia de 2 % entre los resultados más altos y más bajos de precisión, tanto con el conjunto de entrenamiento como con el conjunto de prueba. El modelo en MLP obtuvo 98 % de precisión, el modelo SVC 97 %, y el modelo LR obtuvo 96 %. Por otra parte, con el conjunto de prueba, SVC obtuvo un 1 % más que el MLP y 2 % más que el LR, no obstante, como se puede corroborar en la matriz de confusión, los modelos MLP y LR tuvieron menos casos incorrectos —cinco casos cada uno— mientras que SVC clasificó incorrectamente seis documentos.

	Precisión		
	MLP	SVC	LR
Entrenamiento	0.9859	0.9795	0.9617
Prueba	0.9416	0.9508	0.9340

Tabla 6.2 – Promedio de precisión por modelo de clasificación en catalán.

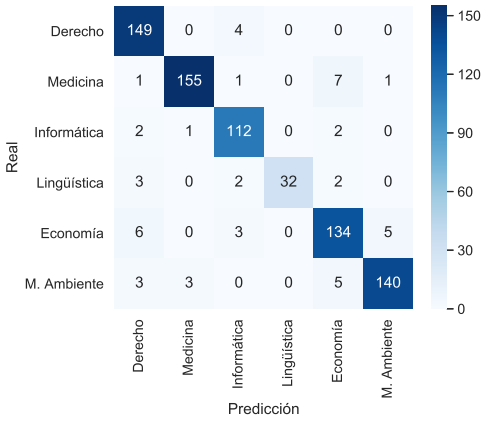


Figura 6.5 – Evaluación de predicciones de tema del modelo LR en catalán.

6.2.2 Resultados de modelos de clasificación en español

Tras el entrenamiento del modelo SVC en español la precisión promedio que se obtuvo con el subconjunto de entrenamiento fue igual a 98 % y, tras su evaluación, el subconjunto de prueba también obtuvo 98 % de precisión promedio. En el diagrama 6.6 podemos observar que hubo un total de 237 documentos clasificados correctamente como pertenecientes a *informática* y solamente un caso incorrecto clasificado como *economía*.

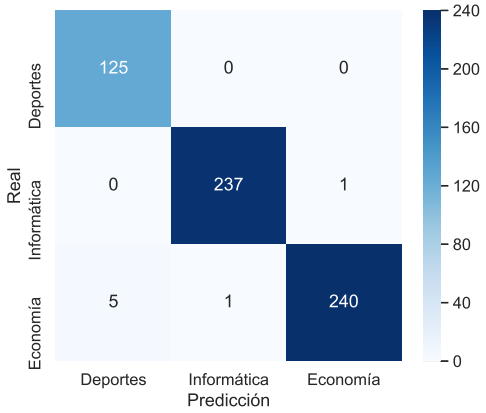


Figura 6.6 – Evaluación de predicciones de tema del modelo SVC en español.

En modelos basado en MLP obtuvo un total de 98 % de precisión, tanto con el subconjunto de prueba como con el subconjunto de entrenamiento. Este modelo clasificó correctamente la totalidad de los documentos (ver 6.7) del total de 238 documentos pertenecientes que tratan sobre informática, solamente uno ha sido etiquetado como *economía*. Podemos ver el mismo comportamiento en la categoría *economía* donde solo un documento fue etiquetado con el tema *informática*.

El modelo restante, LR, obtuvo resultados similares a los dos modelos anteriores: 98 % de precisión promedio en ambos subconjuntos del corpus especializado. La matriz 6.8 nos muestra 236 documentos clasificados de forma correcta con la etiqueta *informática* y solo un caso de error con la etiqueta *economía*.

En la tabla 6.3 se muestran los resultados obtenidos por todos los modelos evalua-

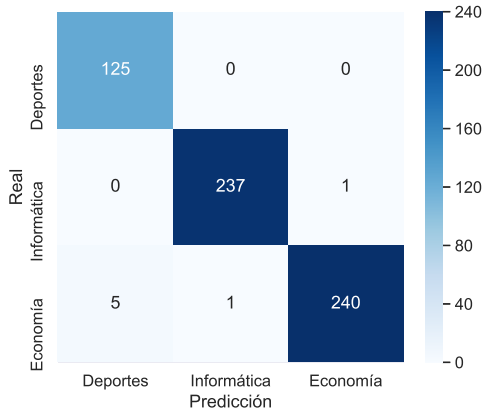


Figura 6.7 – Evaluación de predicciones de tema del modelo MLP en español.

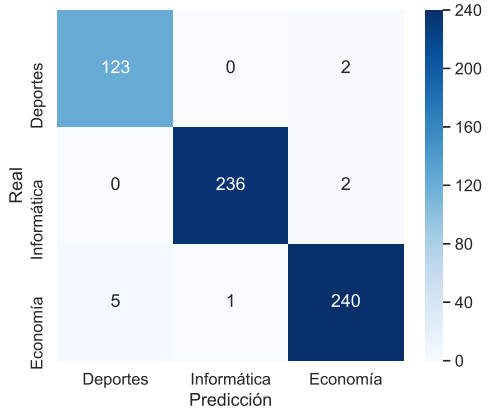


Figura 6.8 – Evaluación de predicciones de tema del modelo LR en español.

dos. Todos obtuvieron porcentajes muy similares tanto en entrenamiento como en prueba. Mientras que no hubo variación en los resultados del conjunto de entrenamiento, en el conjunto de prueba podemos ver una pequeña variación entre cada metodología. Como regla general, un resultado perfecto suele ser indicador de un modelo sobreajustado o de una clase sobreajustada, podemos observar este comportamiento en el modelo MLP.

	Precisión		
	MLP	SVC	LR
Entrenamiento	0.9862	0.9862	0.9862
Prueba	0.9868	0.9885	0.9835

Tabla 6.3 – Promedio de precisión por modelo de clasificación en español.

6.2.3 Resultados de modelos de clasificación en francés

El primer modelo evaluado fue el SVC, dicho modelo obtuvo 97 % de precisión con el conjunto de entrenamiento y 91 % con el conjunto de prueba. Estos valores equivalen a un total de 246 (ver 6.9) casos clasificados correctamente como documentos con contenido

relacionado con la informática y 25 fueron clasificados incorrectamente. De los 25 casos clasificados incorrectamente, 22 fueron etiquetados como *economía* y 3 como *deportes*.

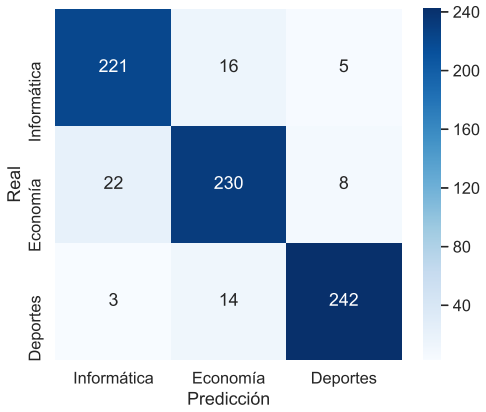


Figura 6.9 – Evaluación de predicciones de tema del modelo SVC en francés.

Por otra parte, el modelo MLP (ver figura 6.10) obtuvo 97 % de precisión con el conjunto de entrenamiento y 90 % con el conjunto de prueba. En la matriz de confusión podemos comprobar que hubo un total de 225 documentos clasificados correctamente con la etiqueta *informática*. Por otra parte, se registraron 27 documentos clasificados incorrectamente como *economía* y tres documentos clasificados como *deportes*.

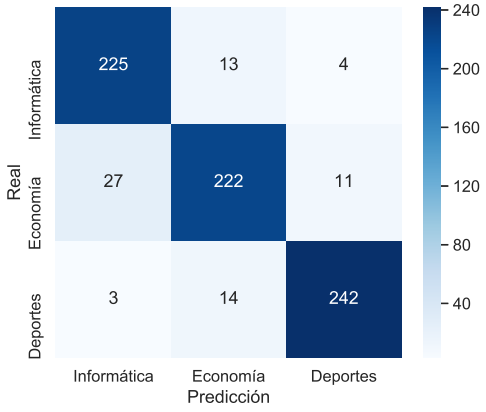


Figura 6.10 – Evaluación de predicciones de tema del modelo MLP en francés.

El modelo LR generó los siguientes resultados: 96 % de precisión en el conjunto de entrenamiento y 91 % con el conjunto de prueba. Estos valores podemos interpretarlos mediante la matriz de confusión 6.11, en la cual hay un total de 220 documentos clasificados correctamente y 20 incorrectamente, 18 fueron clasificados como *economía* y dos como *deportes*.

Entre el promedio de precisiones (ver tabla 6.4) obtenidas con el subconjunto de entrenamiento hubo una variación de 1 %, siendo el modelo LR el modelo con el resultado más bajo: 96 % frente al 97 % de los modelos MLP y SVC. La precisión promedio obtenida con el subconjunto de prueba también tuvo una variación de 1 %, el modelo MLP obtuvo 90 % y los modelos SVC y LR 91 %.

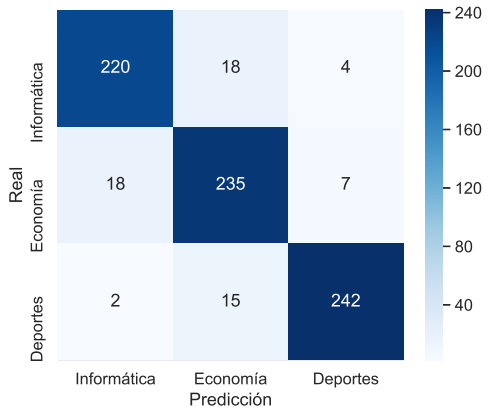


Figura 6.11 – Evaluación de predicciones de tema del modelo LR en francés.

	Precisión		
	MLP	SVC	LR
Entrenamiento	0.9740	0.9740	0.9643
Prueba	0.9053	0.9106	0.9159

Tabla 6.4 – Promedio de precisión por modelo de clasificación en francés.

6.2.4 Selección de modelo para implementación final: Regresión Logística

En vista de los resultados obtenidos en las secciones anteriores decidimos utilizar un modelo LR para nuestro sistema. Esta selección se encuentra motivada, principalmente, por las siguientes razones:

- Simplicidad del modelo: MLP y SVC son modelos más complejos en comparación con los basados en LR. Un modelo simple puede ser entrenado en un menor tiempo, requiere menos recursos y realiza su tarea con mayor rapidez.
- Capacidad de generalización: Las diferencias entre porcentajes de precisión no son significativas una vez que se analizan los documentos que fueron clasificados por cada modelo. A pesar de obtener porcentajes más bajos, los modelos LR mostraron una capacidad de generalización similar al resto de modelos evaluados. Aplicamos así, la intuición de la navaja de Ockham.

Para tener una idea de la capacidad de generalización de los modelos LR en español y catalán, empleamos como datos de entrada todos los contextos obtenidos de la base de datos de OBNEO que fueron seleccionados manualmente. A cada contexto le fue asignada la etiqueta *informática* y, posteriormente, se introdujeron al clasificador para comprobar que el modelo es capaz de clasificarlos correctamente.

Las condiciones anteriores implican que solamente hay una posible clase que puede ser asignada, *informática* en ambos casos. La intención de este experimento es validar que nuestros modelos pueden clasificar correctamente los contextos que fueron seleccionados manualmente. Los resultados de este experimento se pueden observar en la tabla 6.5.

	Precisión	Exhaustividad	f1-Score	Soporte
Catalán	1.0	0.42	0.60	120
Español	1.0	0.28	0.44	194

Tabla 6.5 – Resultados de clasificación temática de las concordancias del OBNEO.

En ambos idiomas se obtuvo un valor del 100 % de precisión, es decir que de todos los documentos analizados por el sistema, todos fueron relevantes. Este resultado no es significativo ya que se trata de documentos que pertenecen a una única clase. Los porcentajes de exhaustividad fueron iguales a 41 % en catalán y 28 % en español, este valor nos indica la cantidad de positivos reales obtenidos por cada modelo.

El *f1-score* toma en cuenta el resultado obtenido en precisión y exhaustividad para calcular el promedio ponderado por clase. En catalán obtuvo un valor de 0.58 y en español 0.44, que nos indican cuán robustos son los modelos, es decir, la capacidad de generalización o de realizar predicciones de forma precisa y correcta.

Mientras que los resultados no fueron tan altos como con los datos de entrenamiento, estos son resultados con contextos distintos a los usados para el entrenamiento del sistema y, además, son contextos cortos. Cabe destacar esta segunda característica ya que la longitud promedio de los contextos empleados es de 23.32 palabras en catalán y 26.64 palabras en español.

6.3 Extracción de palabras claves con etiquetas gramaticales

Evaluamos la extracción de palabras claves (KW) adaptando el método propuesto por Mihalcea y Tarau (2004). Esta metodología de evaluación consistió en emplear un corpus de resúmenes académicos acompañados de las palabras clave que los describen para comprobar el porcentaje de casos correctos que el algoritmo es capaz de extraer.

Nuestra implementación de TextRank genera como resultado una lista de palabras en texto simple. Estas palabras, tal como se ha mencionado en la sección 5.6 han sido filtradas para limitar los resultados por adjetivos, sustantivos y verbos. En nuestro caso, el proceso de comprobación consiste en verificar que, en la lista generada por nuestra implementación, aparezca el candidato a NS que ha sido previamente detectado.

Cabe mencionar que la evaluación se llevará a cabo con formas y no con lemas, esta decisión imita el proceso de detección manual. En un entorno de trabajo manual, los procesos de normalización y lematización se realizan una vez que se han extraído las formas y se analiza su variación ortográfica, para seleccionar la adecuada a cada caso.

Durante la evaluación llevamos a cabo un proceso de comprobación doble: realizamos una prueba de extracción empleando lemas y una segunda prueba de extracción empleando formas. Puesto que no todas las formas y lemas de los candidatos a NS detectados son iguales, esperamos tener siempre un número mayor de formas en comparación con los lemas. En el caso de que esta condición no se cumpla, podríamos inferir que el algoritmo no está funcionando de manera adecuada.

En la tabla 6.6 se muestran los totales obtenidos de este proceso de evaluación. La

cantidad de formas detectadas en catalán fue de 42 de 120 candidatos. Esta cifra equivale a 35.00 % de unidades extraídas correctamente. Por otra parte, el análisis por lemas en catalán resultó de 36 de 120 candidatos que representan un total del 30.00 % del total. Con el corpus en español se obtuvieron resultados similares. En cuanto a formas, se recuperaron 70 de 194, que representan el 36.08 % del total. Y, finalmente, con respecto a los lemas, se extrajeron un total de 69 de 194 que equivale a 35.57 % del total esperado.

	Casos	Recuperado	Porcentaje
Formas CA	120	42	35.00 %
Lemas CA	120	36	30.00 %
Formas ES	194	70	36.08 %
Lemas ES	194	69	35.57 %

Tabla 6.6 – Total de palabras claves obtenidas con TextRank por formas y lemas en catalán y español.

Los porcentajes de formas y lemas en español tuvieron una variación del 1 %, mientras que en catalán hubo una variación del 5 %. A pesar de que estos resultados no son ideales, tanto en catalán como en español debemos tener en cuenta el factor de la longitud de las concordancias ya que, como se menciona en la sección 5.6, este algoritmo funciona con un sistema de votación. Cada concordancia tiene una longitud promedio de 23.32 palabras en catalán y 26.64 palabras en español. Esta longitud tiene un impacto en los resultados de la extracción.

La finalidad del registro de contextos de aparición en la base de datos del OBNEO es llevar un registro del uso de cada neologismo detectado. Como sugerencia de mejora a futuro para la plataforma OBNEO, podría proponerse el registro del párrafo donde aparece el candidato a neologismo. Un contexto más extenso e informativo permite detectar con mayor facilidad las unidades de interés.

6.4 Generación de campos semánticos mediante representaciones vectoriales de palabras

Como se mencionó en el capítulo 5, los resultados obtenidos de la evaluación del primer acercamiento indican que el uso de diccionarios como fuente para la representación conceptual de una palabra, tiene como desventaja la generación de representaciones parciales y no útiles. Para dar respuesta a esta carencia hemos optado por la generación de campos semánticos (CS) mediante el uso de *word embeddings*. Estos CS están compuestos por las palabras más similares a una palabra consultada, donde dicha similitud es calculada mediante la función de coseno euclidiano. Bajo este supuesto, cada palabra contará con un campo semántico que de cuenta de su significado prototípico —o más frecuente— en la lengua general.

La primera etapa de análisis consiste en comprobar la efectividad de estos campos semánticos como sustituto de las definiciones del diccionario. Para este fin generamos dos conjuntos de datos por lengua de trabajo un conjunto de lemas y un conjunto de formas. Se han generado estos dos conjuntos (ver tabla 6.7) puesto que en la base de

datos de OBNEO se registran los lemas normalizados de los candidatos a NS, pero en las concordancias que se se registran se conservan las formas. En total se han obtenido el siguiente número de formas y lemas:

	Casos	Formas	Lemas
Catalán	120	87	77
Español	194	140	128

Tabla 6.7 – Relación de lemas y formas totales en catalán y español.

La evaluación de los CS se tratará como un problema de clasificación binaria, donde las dos clases a ser detectadas serán *informática* y *otro*. Por una parte se asignará manualmente la temática real que se observa en cada CS y, posteriormente, se usará el modelo de clasificación por LR para detectar automáticamente la temática de cada CS generado. Por lo tanto, se evaluarán los CS obtenidos de los tres modelos de *embeddings* Sense2Vec (S2V), FastText (FT) y Word2Vec (W2V) mencionados en la sección 5.7.

La tercera etapa de evaluación consistirá en valorar la capacidad de desambiguación de significado empleando los CS —como sustitutos de las fuentes lexicográficas de referencia habitualmente usadas— y las concordancias del OBNEO. Partiendo de la idea de que un candidato a NS puede ser encontrado dentro de un contexto distinto a su contexto temático habitual, el contexto novedoso está representado por la concordancia obtenida de la base de datos del OBNEO y el contexto habitual por el CS.

Durante esta etapa final, tomaremos las KW que se obtuvieron con TextRank acompañadas de la temática que les ha sido asignada automáticamente y generaremos un CS por KW para comparar si existe concordancia entre temáticas. Posteriormente se repetirá este proceso con todas las unidades que no fueron recogidas por TextRank ya que, a pesar de no han sido detectadas por el sistema, nos servirán como indicador del carácter neológico de los NS registrados en la base de datos del OBNEO.

Este proceso de validación podrá tener tres posibles escenarios. En el primero las unidades que no se encuentren dentro del vocabulario del modelo de *embeddings* que está siendo evaluado se presentarán como candidatos a neologismo de forma. Por otra parte, las unidades cuyo CS y concordancia tengan asignada la misma temática no serán considerados candidatos a NS, ya que el contexto da cuenta de la temática predominante en la lengua general representada por el CS. Finalmente, cuando no exista concordancia entre temas de una unidad, el sistema presentará dicha unidad como candidato a NS.

6.4.1 Embeddings y campos semánticos obtenidos con Word2Vec

Del total de 194 concordancias se obtienen 140 formas y 128 lemas en español. El modelo Word2Vec (W2V) en español generó representaciones para 120 de las 140 formas, esto se traduce en 20 elementos que no se encuentran dentro del vocabulario del modelo. Tras generar los CS de cada forma y analizar su temática manualmente, se realizó la evaluación de la clasificación automática empleando las métricas usuales: *f1-score*, precisión, exhaustividad y soporte.

En la tabla 6.8 podemos observar que la clasificación de los campos semánticos correspondientes a la informática dio mejores resultados que aquellos pertenecientes a otras

temáticas. El *f1-score* obtenido por la categoría informática es igual a 91 %, esta cifra es 4 % superior al 87 % obtenido por otro. Esta variación muestra que el clasificador ha logrado asignar correctamente el tema *informática* a aquellos CS que, en efecto, pertenecer a esta categoría. En cambio, el resultado obtenido por la categoría *otro* indica que el modelo clasificador tiene más problemas con los contextos que corresponden a otras categorías.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9154	0.9701	0.8666	75.0
Otro	0.8775	0.8113	0.9555	45.0
Micro Media	0.9000	0.9000	0.9000	120.0
Macro Media	0.8965	0.8907	0.9111	120.0
Media Ponderada	0.9012	0.9105	0.9000	120.0

Tabla 6.8 – Detección de temas de CS por forma generados con Word2Vec en español.

A pesar de esta variación, la media ponderada de todas las métricas de evaluación se encuentra sobre el 90 %, estos valores se acercan al 91 % de precisión promedio obtenido durante la evaluación del modelo clasificador con LR con el conjunto de prueba. Estos resultados se encuentran dentro del margen esperado, no obstante hay una variación significativa entre los valores de precisión ya que *otro* obtuvo un 81 % de precisión, mientras que *informática* 97 %, en total 16 % de variación entre clases.

Por otra parte, del total de 128 lemas en español se obtuvieron 113 CS. En la tabla 6.9 se muestra la media ponderada de las métricas, en todos los casos se encuentra alrededor del 87 %, una diferencia de 3 % en comparación con los resultados obtenidos por formas en español. La variación entre resultados ha sido menor, siendo de nuevo la *precisión* la métrica que presenta mayor variación entre clases: 91 % en *informática* y 82 % en *otro*.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.8985	0.9117	0.8857	70.0
Otro	0.8409	0.8222	0.8604	43.0
Micro Media	0.8761	0.8761	0.8761	113.0
Macro Media	0.8697	0.8669	0.8730	113.0
Media Ponderada	0.8766	0.8776	0.8761	113.0

Tabla 6.9 – Detección de temas de CS por lema generados con Word2Vec en español.

En cuanto a las formas obtenidas en catalán, se cuenta con 120 concordancias o casos, de los cuales se han obtenido 87 formas. El modelo W2V en catalán ha generado CS para 60 de 87 formas. En la tabla 6.10 podemos observar que los resultados del *f1-score*, en ambas clases, se encuentran por encima del 90 %, *otro* obtuvo 91 % mientras que *informática* 96 %. No obstante, la métrica exhaustividad presentó una variación del 15 % entre clases, que puede interpretarse como la habilidad del modelo para encontrar todos los casos verdaderos de forma correcta. En este sentido, los CS generados por la categoría *otro* obtuvieron un 85 % en exhaustividad.

Por lo que corresponde a los lemas (ver tabla 6.11), se obtuvieron CS para 58 unidades de un total de 77. De la misma forma que en el caso de los CS generados por forma,

podemos observar una variación entre clases de 18 % en la métrica exhaustividad. No obstante, la media ponderada en todas las métricas se encuentra alrededor del 95 %, siendo la media ponderada de *f1-score* la única con un valor inferior (94 %).

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9677	0.9375	1.0000	45.0
Otro	0.9189	1.0000	0.8500	20.0
Micro Media	0.9538	0.9538	0.9538	65.0
Macro Media	0.9433	0.9687	0.9250	65.0
Media Ponderada	0.9527	0.9567	0.9538	65.0

Tabla 6.10 – Detección de temas de CS por forma generados con Word2Vec en catalán.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9662	0.9347	1.0000	43.0
Otro	0.9032	1.0000	0.8235	17.0
Micro Media	0.9500	0.9500	0.9500	60.0
Macro Media	0.9347	0.9673	0.9117	60.0
Media Ponderada	0.9484	0.9532	0.9500	60.0

Tabla 6.11 – Detección de temas de CS por lema generados con Word2Vec en catalán.

Los resultados obtenidos en todas las métricas evaluadas (tanto en formas como en lemas) en catalán superan a los resultados promedio obtenidos en español, estos resultados son comparables a los porcentajes observados durante la evaluación del modelo clasificador con regresión logística . Sin embargo, la clasificación de CS en español presenta menor variación entre clases. Podemos decir que el modelo W2V en catalán genera mejores CS pertenecientes al campo de la informática en comparación aquellos con CS que pertenecen a otras categorías. Por su parte, los CS generados con el modelo W2V en español fueron clasificados con menor precisión (91 % de precisión en español en comparación con 95 % de precisión en catalán), pero presenta menor desviación entre clases.

6.4.2 Embeddings y campos semánticos obtenidos con FastText

El modelo FastText (FT) generó CS para las 140 formas disponibles en español. Sin embargo, los CS generados con FT no han sido tan informativos como aquellos generados con el modelo Word2Vec (W2V). Podemos comprobar esta carencia observando los resultados de la tabla 6.12. Los valores obtenidos en *f1-score* y exhaustividad se encuentran un 30 % por debajo en comparación a los obtenidos con el modelo W2V. En cuanto a los valores de *f1-score* por clase, *informática* obtuvo 63 % y la clase *otro* 71 %, nuevamente por debajo del 96 % y 90 % obtenidos por el modelo W2V.

A pesar de que los CS generados con FT han obtenido un valor de precisión igual a 97 % para la clase *informática*, el resultado para la clase *otro* ha sido de 56 %, un 41 % de variación entre clases. Observamos, también, una variación similar entre clases en la métrica exhaustividad, donde hubo un resultado de 46 % para *informática* y 98 % para

otro. Estas variaciones han dado como resultado en una media ponderada para *f1-score* de 66 % y 80 % en precisión, encontrándose esta última 10 % por debajo del valor obtenido con el modelo W2V.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.6341	0.9750	0.4698	83.0
Otro	0.7133	0.5600	0.9824	57.0
Micro Media	0.6785	0.6785	0.6785	140.0
Macro Media	0.6737	0.7675	0.7261	140.0
Media Ponderada	0.6664	0.8060	0.6785	140.0

Tabla 6.12 – Detección de temas de CS por forma generados con FastText en español.

En cuanto a los lemas en español, el modelo FT ha generado 128 CS para los 128 lemas disponibles, no obstante en la tabla 6.13 observamos que se repite el mismo comportamiento observado en los CS obtenidos por formas en español: alto índice de *precisión* en la clase *informática* (97 %) y bajo para la clase *otro* (51 %); y bajo índice de exhaustividad para la clase *informática* (43 %) y alto para la clase *otro* (97 %). En el primer caso hay una variación 46 % entre clases y el segundo caso una variación de 54 % entre clases.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.6034	0.9722	0.4375	80.0
Otro	0.6714	0.5108	0.9791	48.0
Micro Media	0.6406	0.6406	0.6406	128.0
Macro Media	0.6374	0.7415	0.7083	128.0
Media Ponderada	0.6289	0.7992	0.6406	128.0

Tabla 6.13 – Detección de temas de CS por lema generados con FastText en español.

El valor máximo obtenido en medias ponderadas fue obtenido en la métrica precisión, con un 79 %, mientras que los resultados para *f1-score* y exhaustividad fueron de 62 % y 64 % respectivamente. A pesar de que el valor máximo obtenido ha sido en precisión, dicha métrica se encuentra 8 % por debajo del valor observado en el modelo W2V para lemas en español y el *f1-score* también es inferior al obtenido con el modelo W2V por una diferencia de 25 %.

Por lo que corresponde a las formas en catalán, de nuevo el modelo FT ha sido capaz de generar CS para las 87 formas disponibles. En comparación con el modelo equivalente en español, podemos observar una mejoría en la capacidad de generalización, ya que en este caso se observa menor variación entre categorías en todas las métricas y las medias ponderadas son superiores: *f1-score* y exhaustividad obtuvieron un 79 % respectivamente, mientras que en precisión un 80 %. Estos resultados siguen siendo cifras inferiores a las obtenidas con el modelo W2V en catalán, encontrándose un 16 % por debajo en las métricas *f1-score* y exhaustividad, y un 15 % por debajo en precisión.

Finalmente, con la implementación del modelo FT se han podido obtener CS para los 77 posibles lemas en catalán. A diferencia del comportamiento observado en las formas en catalán, en este caso existe una variación considerable entre clases para las métricas

exhaustividad y precisión. El modelo de LR ha sido capaz de clasificar con 97 % de precisión aquellos lemas con CS pertenecientes a la informática (ver tabla 6.15), pero, por otra parte, los CS pertenecientes a otras temáticas han obtenido una precisión igual a 51 %. En cuando a exhaustividad observamos un comportamiento similar, la categoría *informática* obtuvo un 43 % y la categoría *otro* un 97 %.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.7954	0.7446	0.8536	41.0
Otro	0.7906	0.8500	0.7391	46.0
Micro Media	0.7931	0.7931	0.7931	87.0
Macro Media	0.7930	0.7973	0.7963	87.0
Media Ponderada	0.7929	0.8003	0.7931	87.0

Tabla 6.14 – Detección de temas de CS por forma generados con FastText en catalán.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.6034	0.9722	0.4375	80.0
Otro	0.6714	0.5108	0.9791	48.0
Micro Media	0.6406	0.6406	0.6406	128.0
Macro Media	0.6374	0.7415	0.7083	128.0
Media Ponderada	0.6289	0.7992	0.6406	128.0

Tabla 6.15 – Detección de temas de CS por lema generados con FastText en catalán.

Estas variaciones entre métricas indican que el modelo clasificador ha tenido mayor dificultad para clasificar correctamente los casos verdaderos positivos. Esto se ve reflejado en el resultado por media ponderada del *f1-score*, donde se hubo solo 62 %. A pesar de que la media ponderada de precisión es de 79 %, esta cifra es un 16 % inferior al 95 % obtenido por lemas en catalán observado en los CS generados con el modelo W2V.

6.4.3 Embeddings y campos semánticos obtenidos con Sense2Vec

El modelo Sense2Vec (S2V) permite generar representaciones vectoriales de palabras con desambiguación por etiquetas gramaticales. En total se han obtenido 109 CS de 140 posibles formas en español. Las consultas para generar cada CS se realizaron en conjunto con las etiquetas gramaticales que se encuentran registradas en la base de datos. Como se puede ver en la tabla 6.16, los resultados de todas las métricas empleadas se encuentran dentro de un rango similar a las cifras obtenidas con los CS generados con W2V. La categoría *otro* presenta una mejora en comparación con el modelo W2V: 90 % en *f1-score*, 85 % en precisión y 96 % en exhaustividad. Por otra parte, *informática* tuvo una variación negativa del 1 % en las métricas *f1-score* y precisión. Estos resultados pueden ser evidencia de mejora en la informatividad de los CS que han sido generados.

En cuanto a los lemas (ver tabla 6.17), S2V generó 101 CS del total de 128. En este caso las medias ponderas de todas las métricas son superiores a las obtenidas en la evaluación del modelo de CS por lemas generado con W2V, en todos los casos hubo un

incremento del 3 % para dar un total de 90 % en comparación con el 87 % obtenido por el modelo equivalente. Este incremento puede haber ocurrido gracias a que las etiquetas gramaticales permiten una organización conceptual de las representaciones distinta a las representaciones generadas por el modelo W2V, ya que ambos modelos han sido entrenados con una arquitectura Skip-Gram empleando los mismos corpus.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9090	0.9615	0.8620	58.0
Otro	0.9074	0.8596	0.9607	51.0
Micro Media	0.9082	0.9082	0.9082	109.0
Macro Media	0.9082	0.9105	0.9114	109.0
Media Ponderada	0.9083	0.9138	0.9082	109.0

Tabla 6.16 – Detección de temas de CS por forma generados con Sense2Vec en español.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9038	0.9400	0.8703	54.0
Otro	0.8979	0.8627	0.9361	47.0
Micro Media	0.9009	0.9009	0.9009	101.0
Macro Media	0.9009	0.9013	0.9032	101.0
Media Ponderada	0.9011	0.9040	0.9009	101.0

Tabla 6.17 – Detección de temas de CS por lema generados con Sense2Vec en español.

En catalán fueron generados CS para 64 de 88 formas, que es la misma cifra conseguida con el modelo W2V. En cuanto a los resultados de la clasificación, estos han mantenido una media ponderada del 98 % en todas las métricas analizadas, el valor máximo observado en todos los casos. Este resultado podría deberse a que el modelo de clasificación entrenado en catalán también mostró resultados superiores en comparación con las otras lenguas. Debemos recordar que, a pesar de que los *embeddings* son generados por un modelo neuronal, para la clasificación detección de temas optamos por emplear un modelo más simple basado en LR.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9866	1.0000	0.9736	38.0
Otro	0.9811	0.9629	1.0000	26.0
Micro Media	0.9843	0.9843	0.9843	64.0
Macro Media	0.9838	0.9814	0.9868	64.0
Media Ponderada	0.9844	0.9849	0.9843	64.0

Tabla 6.18 – Detección de temas de CS por forma generados con Sense2Vec en catalán.

Finalmente, en el caso de los lemas, se han obtenido 58 CS de los 77 esperados, sin embargo, de la misma manera que en el caso de las formas, los resultados de todas las métricas rondan el 98 %. Así pues podemos concluir que el modelo S2V ha generado los CS

más informativos a pesar de haber producido una cantidad menor de CS en comparación con los modelos FT y W2V. Cabe señalar que, a pesar de que una lista de pares ordenados (tupla) unidad - etiqueta consultada no cuente con una representación, existe la probabilidad que dicha unidad cuente con una representación bajo otra etiqueta gramatical. Esto se debe a que durante el proceso de etiquetado, le ha sido asignada una etiqueta distinta a la esperada de forma automática.

	f1-Score	Precisión	Exhaustividad	Soporte
Informática	0.9850	1.0000	0.9705	34.0
Otro	0.9795	0.9600	1.0000	24.0
Micro Media	0.9827	0.9827	0.9827	58.0
Macro Media	0.9823	0.9800	0.9852	58.0
Media Ponderada	0.9828	0.9834	0.9827	58.0

Tabla 6.19 – Detección de temas de CS por lema generados con Sense2Vec en catalán.

En conclusión, los mejores resultados en ambas lenguas fueron obtenidos con la generación de CS empleando S2V. A diferencia del modelo FT y W2V, este modelo emplea etiquetado gramatical para generar representaciones vectoriales de palabras que cuentan con información más detallada. Los modelos W2V y S2V fueron entrenados en condiciones similares con los mismos datos de entrenamiento y bajo la misma arquitectura y, a pesar de generar una menor cantidad de CS, los resultados de esta evaluación se encuentran a la par de los observados en la sección 6.2.

6.4.4 Desambiguación de significado mediante representaciones vectoriales de palabras y clasificación automática

El proceso de desambiguación de significado se lleva a cabo evaluando la concordancia de temas entre el significado principal en la lengua general de un candidato y el tema de la concordancia que está siendo evaluada. En la sección anterior evaluamos la pertinencia del uso de un CS compuesto por las palabras más similares a una palabra consultada. En la tabla 6.20 podemos ver un resumen de los resultados de dicha evaluación. Así, tomando la métrica *f1-score* como referencia, optamos por emplear S2V para la generación de campos semánticos ya que dicho modelo generó los CS que fueron clasificados automáticamente con mayor éxito, tanto para formas como para lemas.

El proceso para desambiguar los nuevos significados para obtener candidatos a NS consiste en comparar la temática de la concordancia de cada KW detectada en el texto que está siendo analizado, contra el CS de dicha KW, en el supuesto de que dicha palabra existe dentro del vocabulario del modelo neuronal. Bajo esta premisa podemos encontrar los siguientes tres escenarios:

1. La unidad consultada no se encuentra en el modelo.

2. La concordancia y el CS tienen la misma temática.

3. La concordancia y el CS tienen temáticas diferentes.

Modelo	Formas	% Formas	Lemas	% Lemas	f1-Formas	f1-Lemas
			Catalán			
W2V	60/87	68.96 %	58/77	75.32 %	0.9527	0.9484
FT	87/87	100 %	77/77	100 %	0.7929	0.6289
S2V	64/87	73.56 %	58/77	75.32 %	**0.9844**	**0.9828**
			Español			
W2V	120/140	85.71 %	113/128	88.28 %	0.9012	0.8766
FT	140/140	100 %	128/128	100 %	0.6664	0.6289
S2V	109/140	77.85 %	101/128	78.90 %	**0.9083**	**0.9011**

Tabla 6.20 – Resumen de resultados de CS de formas por modelo.

En el escenario 1, el sistema presenta la palabra a candidato a neologismo formal, de forma similar que se realizaría siguiendo el criterio lexicográfico convencional. En el escenario 2 la KW se descarta ya que la concordancia de temas indica un uso no neológico. El tercer escenario ocurre cuando la temática de la concordancia y el CS son diferentes: en caso de que la concordancia tenga como tema *informática* y el CS *otro*, la KW que está siendo evaluada se presenta como candidato a NS. Por otra parte, cuando la temática de la concordancia es *otro* y la temática del CS es *informática* el candidato es presentado como ambiguo para ser evaluado manualmente.

Para comprobar los candidatos que pueden ser obtenidos mediante esta metodología emplearemos el listado de NS y concordancias del OBNEO. A cada concordancia le ha sido asignado un tema de forma manual y un tema de forma automática (sección 6.2), así mismo, de cada concordancia se extrajeron KW empleando TextRank (sección 6.3). Así pues, por cada contexto y palabra clave candidato se evaluarán los escenarios mencionados para comprobar qué unidades serían evaluadas como candidatos a NS. Por lo tanto, como datos de entrada usamos 120 concordancias en catalán y 194 concordancias en español y realizamos el análisis por formas, ya que, a pesar de que se registran los lemas dentro de la base de datos, las formas son los elementos que se extraen y detectan en los textos que se analizan.

Durante la primera etapa de análisis generamos dos subconjuntos de datos. El primer subconjunto incluye solamente aquellas unidades que no se encontraban registradas dentro del vocabulario del modelo S2V y, el segundo, el resto de unidades que sí cuentan con un CS y una representación vectorial. Comenzando por el primer subconjunto, analizamos por separado las unidades que además de no tener un CS, no fueron detectadas por TextRank[1], los resultados se muestran la tabla 6.21:

Lengua	Detectadas	No detectadas	Total
Catalán	7	22	29
Español	8	29	37

Tabla 6.21 – Unidades con CS nulo.

[1]Debemos recalcar la distinción entre candidato y caso, un candidato a NS puede tener múltiples casos y haber sido detectado por TextRank solamente en una de sus múltiples concordancias o casos.

En catalán las formas *correus electrònics, piulada, play 3, play station, y powerpoints* fueron extraídas por TextRank, pero no cuentan un CS. En el caso de *correus electrònics* y *play 3* y *play station*, son unidades que se esperaba que no se encontrasen dentro del modelo ya que fue entrenado con unigramas, esta condición excluye del análisis a las unidades polilexemáticas. En español el modelo excluyó las siguientes unidades: *gameboys, ipod, play station, powerpoint, quake, tamagotchis y watsap*.

En cuanto a las unidades que no fueron detectadas por TextRank y que no cuentan con un CS encontramos 22 casos con 18 candidatos únicos en catalán y 29 casos con 25 candidatos únicos en español que se encuentran listados a continuación:

> Catalán: *adobe photoshop, blue ray, chat roulette, descarregar-se, disc dur, iphones, internets, microsoft word, piulades, piuladors, piulaire, piular, piularlos, play station III, playstations, power point, tuitters, whatsapps.*
>
> Español: *cargarse, clonaban, correos electrónicos, cubesat, disco duro, e-mail, famicom, game boy, hotmail, instagrams, ipod, iseries, jaquea, jaquear, macintosh, play-station, power point, powerpoint, skype, tamagochis, tetrix, trojanos, tuenti, webby, whatsapp.*

De nuevo encontramos unidades polilexemáticas que no se encuentran incluidas dentro de los modelos S2V. Sería interesante analizar la pertinencia de considerar neológicos los casos como *marca + producto* (casos como *adobe photoshop*) y *producto + versión* (*play station III*). A pesar de ser composiciones novedosas, en el primer caso se tendría que analizar si el producto puede hace referencia a una entidad sin encontrarse precedido de la marca. En el segundo caso hay evidencia de una secuencia o historial de versiones de un producto, podemos inferir que el objeto *play station* cuenta con, por lo menos, dos versiones anteriores, siendo un candidato ambiguo a NS.

El resto de las unidades fueron detectadas por TextRank y cuentan con CS, es decir que forman parte del vocabulario del modelo S2V. Para desambiguar el significado de los candidatos a NS realizamos dos comparaciones de tema entre concordancias y CS. La primera comparación emplea el tema que fue asignado automáticamente por el clasificador de LR a cada concordancia y el tema del CS que ha sido detectado automáticamente. La segunda utiliza el tema real asignado manualmente a cada candidato (*informática* en todos los casos) y el tema que ha sido detectado por DENISE para cada CS.

En total, el sistema detectó 35 KW en catalán y 62 KW en español (ver tabla 6.22). A partir de estas cifras, el análisis automático de tema presentó 4 contextos con candidatos a NS en catalán y 6 en español. En catalán, DENISE generó los posibles NS *baixar, hiperespai, núvol y photoshop*. Mientras que en español presentó los candidatos *almacenado, dominio, migración, navegabilidad, nube y playstation*. Es interesante que en ambas lenguas se encuentra el candidato *nube* y *núvol* cuyo origen procede de *cloud*, en inglés, un concepto relativamente reciente en comparación con otros candidatos registrados en ambas listas.

Por otra parte, la desambiguación de tema empleando las concordancias etiquetadas manualmente dio como resultado un total de 14 candidatos a NS en catalán y 29 candidatos a NS en español. En el primer caso hubo una variación de 10 candidatos y en el segundo caso una variación de 23 candidatos. Dicha variación puede ser debido a la longitud de las concordancias que fueron analizadas (23.32 palabras en catalán y 26.64

Tema	Posible NS	No neológico	Ambiguo	Total
		Catalán		
Automático	4	13	18	35
Manual	14	21	—	35
		Español		
Automático	6	24	32	62
Manual	29	33	—	62

Tabla 6.22 – Candidatos a NS registrados en el listado obtenido por TextRank.

palabras en español), mientras que los CS tienen una longitud de 150 palabras, esta mayor longitud facilita la tarea del modelo clasificador, mientras que una extensión más corta dificulta su clasificación. No obstante, en ambas lenguas de trabajo los candidatos a NS que surgieron de la clasificación automática de temas se encuentran contenidos en los listados que fueron generados:

> Catalán: *baixar, excel, ferramenta, gurus, hiperespai, mur, núvol, palms, penjades, photoshop, play.*

> Español: *almacenado, api, avatares, bit, blackberry, cuenta, dominio, enlaces, gusano, memes, migración, minis, muro, navegabilidad, navegación, nube, piratería, playstation, soundcloud, tabletas, wikipedia, word.*

En cuanto a los candidatos ambiguos hubo un total de 13 en catalán y 24 en español. Estos listados dan cuenta de aquellos contextos que fueron clasificados incorrectamente con la categoría *otros*, ya que la totalidad debería haber recibido la etiqueta *informática*. Este subconjunto es considerado ambiguo ya que incluye candidatos que, tras analizar los resultados de los contextos analizados manualmente, aparecen tanto en los listados de candidatos a NS, como en el listado no neológico.

Finalmente, también analizamos los casos de candidatos a NS que no fueron detectados durante la etapa de extracción de KW con TextRank. En catalán se registraron un total de 56 unidades que no fueron extraídas, pero que sí se encuentran dentro del léxico de nuestro modelo neuronal. Por otra parte, en español se cuenta con 95 unidades que no fueron detectadas por TextRank, pero que sí cuentan con un CS asociado. Tras el análisis de desambiguación de significado mediante clasificación automática, encontramos un total de 10 candidatos a NS en catalán y 19 en español, estos totales pueden verse en la tabla 6.23.

Los siguientes candidatos a NS fueron resultado de la desambiguación de temas empleando el modelo de clasificación automática tanto en concordancias como en CS:

> Catalán: *antivírics, compte, emule, hostatge, navegació, navegar, núvol, tauleta, tauletes, trols.*

> Español: *alojamiento, androides, apuntadores, asistentes, cablear, caída, clonación, conversión, cortafuego, dominios, identificadores, infectan, motor, nube, palm, telnet, virales, yahoo.*

Tema	Posible NS	No neológico	Ambiguo	Total
		Catalán		
Automático	10	32	14	56
Manual	18	38	—	56
		Español		
Automático	19	41	35	95
Manual	45	50	—	95

Tabla 6.23 – Candidatos a NS excluidos del listado obtenido por TextRank.

Podemos observar que algunos elementos del listado ya han aparecido anteriormente, como es el caso de *núvol* y nube. Esto se debe a que TextRank extrae palabras claves de cada texto analizando los nodos que existen en ese texto y da más importancia a aquellas palabras que son más relevantes mediante un sistema de *voto*. En ambos casos el algoritmo no ha detectado *núvol* ni *nube* como unidades relevantes en el contexto donde han aparecido.

El total de casos ambiguos en catalán ha sido igual a 14, mientras que en español hubo 35 casos. De los 14 casos ambiguos registrados en catalán, 12 fueron clasificados como no neológicos durante la desambiguación empleando las concordancias etiquetadas manualmente. Por otra parte, de los 35 casos ambiguos en español, 24 aparecen dentro del listado no neológico generado con los contextos clasificados manualmente.

En el caso de los candidatos generados tras la desambiguación de tema usando las concordancias anotadas manualmente, se obtuvieron 18 candidatos en catalán, que representan una variación de 8 unidades; mientras que en español se han obtenido 45 candidatos a NS, en total una variación de 26 candidatos. Nuevamente podemos atribuir esta variación a la extensión e informatividad de los contextos, ya que la evaluación de los CS generados con S2V comprobó que la precisión del modelo clasificador es acorde al valor obtenido durante su evaluación. Así, los siguientes candidatos son producto de este análisis:

> Catalán: *antivírics, bussejar, compte, emule, hostatge, mac, murs, navegació, navegar, núvol, penjant, penjar, perfil, pirateria, tauleta, tauletes, trols.*
>
> Español: *agujero, alfombrilla, alojamiento, androides, apuntadores, asistentes, cablear, caerse, cae, caída, clonación, conversión, cortafuego, cuenta, descargaron, dominios, guru, identificadores, infectan, java, mapeo, motor, navegación, navegar, navega, nube, palm, pirateando, play, subir, tamagotchi, telnet, tetris, vacunas, vínculos, virales, virus, visuales, wikipedia, yahoo.*

En ambas lenguas el sistema fue capaz de presentar candidatos sin distinción del nivel de especialización de cada unidad. Todos estos candidatos cuentan con una representación vectorial dentro del modelo S2V y, mediante esta representación, podemos obtener las palabras más similares a un candidato para generar un CS que representa el significado más común de una unidad dentro del vocabulario. Por una parte, la clasificación automática de temas nos ha ilustrado las ventajas y limitaciones de este enfoque y, por

otra parte, el etiquetado manual nos ha permitido ver el funcionamiento de la herramienta en un escenario ideal: conocemos con certeza el tema de cada concordancia y la cantidad de NS que se espera detectar.

Como se ha mostrado, la metodología de DENISE tiene varios elementos que en conjunto pueden servir para detectar NS de forma semiautomática. Los datos que usamos para llevar a cabo las evaluaciones nos han permitido visualizar el tipo de datos de entrada que nuestro sistema usa y las salidas que podemos obtener. No obstante, la extensión de las concordancias a sido una limitante durante el proceso ya que, como se ha visto durante otras etapas de evaluación, es más difícil extraer KW y clasificar concordancias cuando los textos de entrada tienen una extensión limitada.

No obstante, dado que este proceso de evaluación se ha realizado para tener una referencia del funcionamiento de cada etapa del procesamiento del sistema, durante la implementación real del *pipeline* completo podremos comparar el funcionamiento de DENISE una vez desplegado en un contexto de uso práctico. Esta diferenciación es importante puesto que, en un contexto de uso real, los textos de entrada suelen tener una extensión mayor. En vista de los resultados de la evaluación del funcionamiento con contextos cortos, es probable que textos de mayor extensión incrementen la efectividad del sistema.

Capítulo 7

IMPLEMENTACIÓN WEB

Como parte de los objetivos de esta tesis nos hemos planteado el desarrollo de una aplicación de DENISE. Hemos optado por desarrollar una implementación web (ver figura 7.1) ya que, en función en los objetivos de la tesis, una implementación de estas características permite tener un entorno de trabajo multiplataforma y colaborativo, sin necesidad de desarrollar aplicaciones de escritorio para cada sistema operativo.

Figura 7.1 – Página principal de DENISE.

Los componentes claves (descritos en el capítulo 5) han sido desarrollados en su totalidad en Python y, ya que este lenguaje cuenta con múltiples *frameworks* para desarrollar aplicaciones y páginas web. Optamos por usar el *microframework* Flask[1] por su enfoque minimalista e intuitivo. El resto de la interfaz web ha sido desarrollada con Bootstrap[2], para obtener un sitio web con un aspecto moderno: claro, intuitivo y adaptativo.

La página principal de la aplicación cuenta con tres elementos básicos: una barra de navegación, una breve descripción del sistema y un formulario de consulta. La barra de

[1] `http://flask.pocoo.org`
[2] `https://getbootstrap.com`

navegación cuenta con vínculos hacia la página principal mediante los botones *DENISE* e *inicio*, un botón *Acerca de* que lleva al usuario a una página con la información de las instituciones colaboradoras y, finalmente, un botón *Consultas* que lleva al usuario a una página con un formulario para envío de correos electrónicos para realizar consultas sobre el sistema.

En la sección inferior de la figura 7.1 se puede apreciar el campo del formulario donde el usuario puede ingresar un texto para que el sistema lo analice. Este formulario cuenta con los elementos mínimos: un campo en blanco para añadir un texto de hasta 5000 caracteres y un botón *Analizar* que envía el formulario al servidor y da como respuesta una página con el resultado del análisis. La página de resultados (figura 7.2) genera un reporte de cada etapa de procesamiento: detección de tema, extracción de precandidatos, desambiguación de significado y listados de palabras similares.

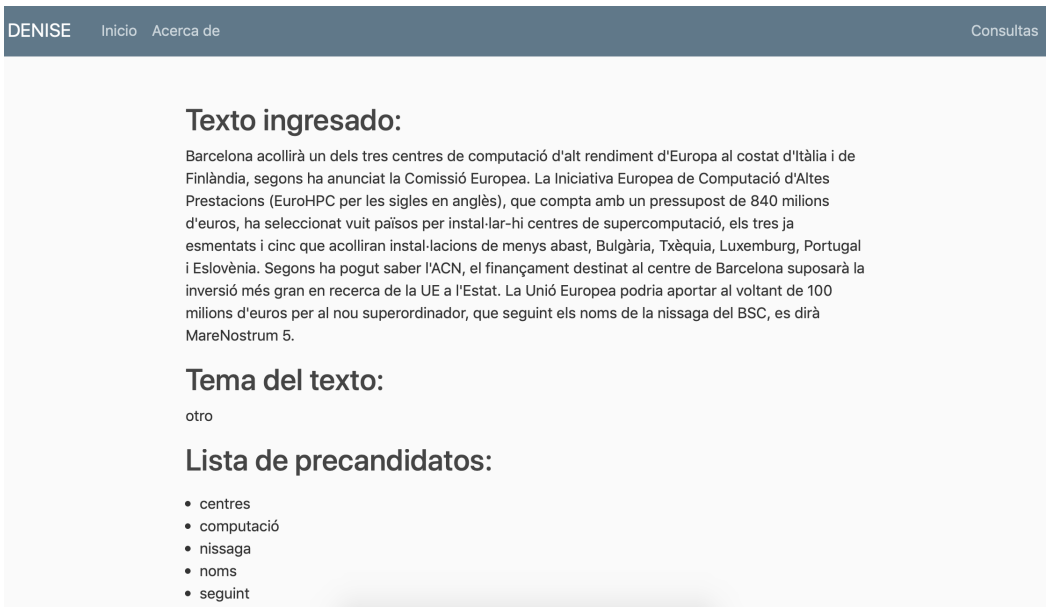


Figura 7.2 – Página de resultados de DENISE.

Los candidatos a NS se presentan en una tabla (ver figura 7.3) que contiene todas las unidades que fueron extraídas con TextRank y que cuentan con una representación vectorial en el modelo Sense2Vec. Cada palabra clave detectada se encuentra acompañada de la temática de su campo semántico, la categoría gramatical que tiene asignada en el modelo neuronal y el resultado de la evaluación de candidato a NS.

Finalmente, el sistema genera un reporte de palabras similares por cada palabra clave que ha sido detectada (ver figura 7.4) en el texto de entrada. Estos listados tienen la finalidad de servir como guía para el usuario decida, finalmente, si alguno de los precandidatos obtenidos es un candidato válido a NS. Este es la característica que define a DENISE como un sistema semiautomático, mientras que el análisis se realiza automáticamente, la selección final es tomada por el usuario.

La página de resultados muestra toda la información del procesamiento para que el usuario tenga herramientas de apoyo para realizar la selección de candidatos. Aunque el diccionario provee la definición de una palabra, los listados de similitud muestran las palabras más comúnmente vinculadas al candidato que está siendo evaluado. Mientras

que estos listados no son una definición lexicográfica, son evidencia de las asociaciones semánticas que una palabra extiende dentro del vocabulario.

Unidades fuera del vocabulario:

Posibles candidatos a neologismo semántico:

Unidad	POS	Tema de embeddings	Estatus
centres	NOUN	otro	no candidato
computació	NOUN	informatica	ambiguo
nissaga	NOUN	otro	no candidato
noms	NOUN	otro	no candidato
seguint	VERB	otro	no candidato

Figura 7.3 – Tabla generada de posibles candidatos a NS.

computació:

Entrada	Similitud	Frecuencia
informàtica	0.7324	711
algorismes	0.6771	508
optimització	0.6719	181
MIT	0.6655	248
ordinadors	0.6632	1829
algoritmes	0.6596	120
modelització	0.6529	48
implementació	0.6517	598
computacionals	0.6323	87
computadors	0.6308	167

Figura 7.4 – Reporte de palabras similares por palabra clave detectada.

En los siguientes apartados presentamos tres casos de uso de la herramienta —uno en cada lengua de trabajo— con contextos de prensa distintos a los empleados durante el proceso de entrenamiento de los modelos de clasificación y, también, distintos a las concordancias de la base de datos del OBNEO. Mediante estos casos de usos podremos observar los resultados que pueden ser esperados, de forma que podamos visualizar el desempeño de la herramienta en un entorno real.

7.1 Caso de uso en español

Para ejemplificar el funcionamiento con contextos reales ingresamos el siguiente texto obtenido del periódico *Clarín*[3]:

[3] `https://clar.in/2IyVQtM`

> “Cerca de una decena de tipos de virus informáticos (malware), diseñados especialmente para infectar smartphones; se empiezan a popularizar entre los ciberdelincuentes por ser herramientas eficaces para lograr que esos dispositivos usen sus procesadores para acuñar criptomonedas sin que el propietario del celular lo detecte, informó una empresa de seguridad informática. Con el auge del bitcoin y otro importante número de monedas virtuales que surgieron en los últimos años, también se comenzaron a diseñar softwares que usan secretamente las capacidades de procesamiento de dispositivos-especialmente servidores y computadoras- para resolver la ecuación matemática con el que ellas se “minan” o acuñan este tipo de divisa virtual.”

Tras ingresar el texto al sistema DENISE (ver figura 7.5), obtuvimos los siguientes precandidatos: *acuñan*, *minan*, *resolver*, *computadoras* y *popularizar*. Este listado se revisa para comprobar que todas las unidades forman parte del vocabulario del modelo neuronal y, en el caso de no encontrarse dentro del vocabulario podrían ser candidatos a neologismos formales. El sistema también detecta que la temática general más probable del texto ingresado es *informática*.

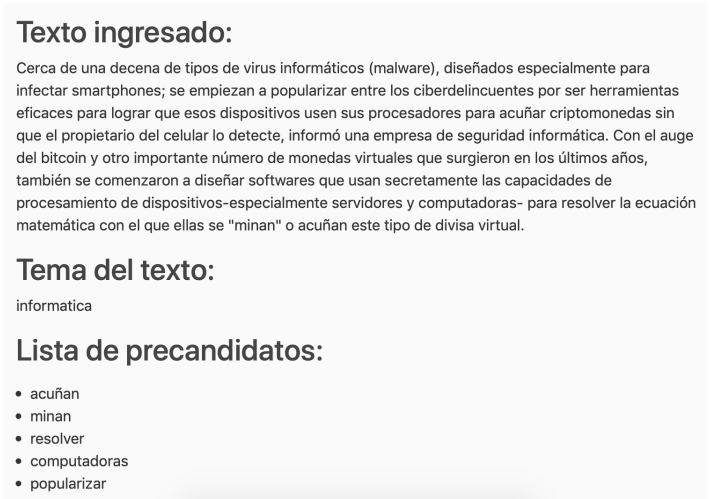


Figura 7.5 – Lista de precandidatos obtenida con DENISE en español.

Posteriormente se analizan las concordancias de temáticas para seleccionar posibles candidatos (figura 7.6). En este caso no hay precandidatos que se encuentren fuera de nuestro vocabulario, por lo tantos todos proceden a ser clasificados. Este proceso descarta *computadoras* como candidato a neologismo semántico ya que DENISE detecta que, tanto la temática del texto ingresado como la temática de su campo semántico pertenece a la categoría *informática*.

En cambio, los precandidatos *acuñan*, *minan*, *resolver* y *popularizar* fueron evaluados como posibles candidatos. Para corroborar esta información, DENISE genera listados de palabras similares, estos listados pueden ser empleados por el usuario para corroborar la evaluación que ha realizado el sistema. Podemos ver en la figura 7.7 que las similitudes de *resolver* y *popularizar* no están siendo empleadas con un significado neológico ya que, en ambos casos, se emplean con sus significados prototípicos en el texto que ha sido analizado.

Unidades fuera del vocabulario:

Posibles candidatos a neologismo semántico:

Unidad	POS	Tema de embeddings	Estatus
acuñan	VERB	otro	candidato
minan	VERB	otro	candidato
resolver	VERB	otro	candidato
computadoras	NOUN	informatica	no candidato
popularizar	VERB	otro	candidato

Figura 7.6 – Lista de posibles candidatos detectados en español.

acuñan:

Entrada	Similitud	Frecuencia
acuñaba	0.6181	50
bimetálicas	0.5938	36
devaluada	0.5858	11
ensayadores	0.5816	6
acuñaron	0.5774	513

(a) Acuñan

minan:

Entrada	Similitud	Frecuencia
vuestros	0.4897	161
descuidan	0.4452	30
conservatorios	0.4392	6
envidiosos	0.4287	47
indignan	0.4253	6

(b) Minan

resolver:

Entrada	Similitud	Frecuencia
solucionar	0.8315	2392
resolverlo	0.706	117
dilucidar	0.6869	354
zanjar	0.6776	111
resolverlos	0.6632	83

(c) Resolver

popularizar:

Entrada	Similitud	Frecuencia
revolucionar	0.6382	169
popularización	0.605	548
masificar	0.5944	45
cimentar	0.5694	208
difundir	0.5655	2821

(d) Popularizar

Figura 7.7 – Listado de palabras similares por candidato a NS en español.

Finalmente, los candidatos *acuñan* y *minan* están siendo empleados con un significado novedoso ya que las similitudes de ambos candidatos aluden a otras temáticas. En el caso de *minan*, en el texto está siendo empleado para referirse al proceso de obtención de criptomonedas, mientras que la similitud de su vector no da cuenta de este significado. De manera similar, *acuñar* también se usa metafóricamente ya que las criptomonedas no pueden ser acuñadas. En este caso la similitud de los embeddings hace referencia al proceso de sellar piezas metálicas para producir monedas mientras que el texto ingresado al proceso informático necesario para generar criptomonedas.

7.2 Caso de uso en catalán

Como caso de uso en catalán empelamos el siguiente fragmento de una noticia publicada en el periódico *El Nacional*[4]:

> "El núvol és el present. Es tracta del processament i arxivament de dades en servidors i va ser creat per a l'usuari a fi de resultar pràctic i fàcil d'utilitzar. Un dels exemples més clars d'aquest servei és el correu electrònic, on els mails s'emmagatzemen en un servidor i no ocupen espai en l'emmagatzemament intern del nostre dispositiu, la qual cosa també ens permet accedir-hi des de qualsevol plataforma. D'altra banda, l'espai que ocupen les fotografies i els vídeos a qualsevol dispositiu és també un pes que el núvol ens treu de sobre. Aplicacions com Google Fotos han permès els usuaris organitzar tot el seu contingut multimèdia als servidors de forma il·limitada, el que representa una gran quantitat d'emmagatzemament lliure al nostre ordinador, tauleta o smartphone."

Durante la primera etapa de análisis DENISE detectó *informática* como el tema principal del texto (ver figura 7.8) y, posteriormente, extrajo los precandidatos siguientes: *servidors*, *dispositiu*, *ocupen*, *núvol* y *exemples*[5].

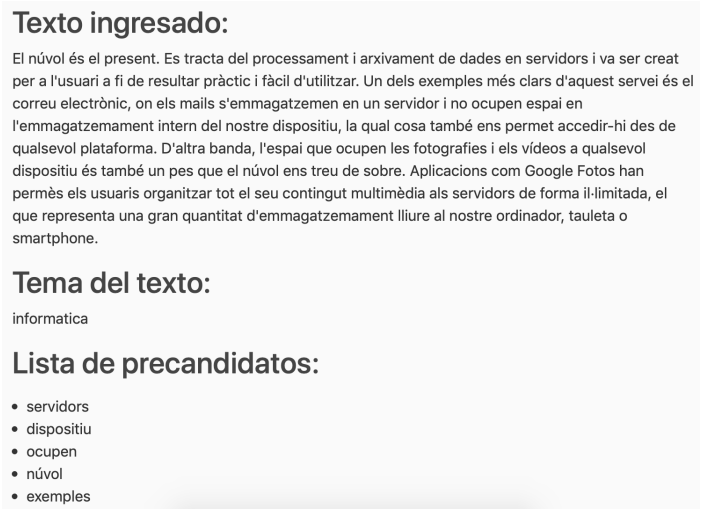


Figura 7.8 – Lista de precandidatos obtenida con DENISE en catalán.

La siguiente etapa de análisis consiste en generar un listado con las palabras más similares a cada precandidato para obtener la temática más común de dicho listado. En dicho listado —o campo semántico— las palabras más relacionadas dan cuenta del significado básico en la lengua general de cada precandidato y, con esta información, el sistema descartó a las unidades *servidor* y *dispositiu* y mantuvo como posibles candidatos a las unidades restantes: *ocupen*, *núvol* y *exemples*.

[4] `https://www.elnacional.cat/ca/tecnologia/nuvol-ordinadors-internet_169596_102.html`

[5] Servidores, dispositivo, ocupen, nube y ejemplos en español.

Posibles candidatos a neologismo semántico:

Unidad	POS	Tema de embeddings	Estatus
servidors	NOUN	informatica	no candidato
dispositiu	NOUN	informatica	no candidato
ocupen	VERB	otro	candidato
núvol	NOUN	otro	candidato
exemples	NOUN	otro	candidato

Figura 7.9 – Posibles a candidatos a NS en catalán.

El sistema procede a mostrar los listados de similitud de cada elemento del listado de posibles candidatos, tanto los que han sido considerados candidatos a neologismo semántico, como a los que no lo han sido. Esta información (ver figura 7.10) ayuda al usuario a seleccionar un verdadero candidato. En el caso de *dispositius* y *servidors*, el significado más común en el modelo de lengua corresponde con un significado prototípico de la informática y, por lo tanto, fueron descartados por esta razón.

dispositiu:

Entrada	Similitud	Frecuencia
perifèric	0.7408	122
ordinador	0.7302	2959
xip	0.7055	348
dispositius	0.7011	1870
maquinari	0.6906	849

(a) Dispositius

exemples:

Entrada	Similitud	Frecuencia
éléments	0.6534	51265
signes	0.6303	11059
termes	0.618	36667
arguments	0.5925	4871
paral·lelismes	0.5917	88

(b) Exemples

núvol:

Entrada	Similitud	Frecuencia
Oort	0.6931	170
núvols	0.666	780
metà	0.6633	515
meteorit	0.6594	162
cúmul	0.6488	567

(c) Núvol

ocupen:

Entrada	Similitud	Frecuencia
ocupan	0.6272	4258
posseeixen	0.584	691
troben	0.572	11714
disposen	0.5714	1014
tenen	0.5616	24413

(d) Ocupen

Figura 7.10 – Listado de palabras similares por candidato a NS en catalán.

Por otra parte, al analizar las palabras más similares a los candidatos *ocupen*, *núvol* y *exemples*, podemos observar que los candidatos *ocupen* y *exemples* han sido clasificados correctamente, pero no están siendo utilizados con un significado novedoso en el texto de entrada, ya que los campos semánticos corresponden a su significado básico. En cambio, el campo semántico de *núvol* hace da cuenta de su significado meteorológico, mientras que en el texto, *núvol*, hace referencia al espacio de almacenamiento de información virtualizado y, por lo tanto, podemos validar *núvol* como un candidato válido a neologismo semántico.

7.3 Caso de uso en francés

Para comprobar el funcionamiento del sistema en francés, utilizamos el siguiente texto obtenido de *Le Figaro*[6]:

> "Quelle tablette choisir? La simple navigation Web et les tâches simples conviennent à une tablette compacte, le multimédia et le jeu demandera une diagonale de 10 pouces pour plus de confort. Une tablette Apple iOS ou Android? Pour certains, c'est presque une histoire de religion, mais le porte-monnaie décidera en premier, les tablettes à la pomme étant plus performantes mais très onéreuses, les homologues Android étant disponibles à des prix très accessibles. Une tablette peut-elle remplacer un PC portable? S'ils sont tous deux capables de diffuser du multimédia et de naviguer sur Internet, un ordinateur sera plus confortable pour de la saisie, du multitâche et des opérations complexes comme l'édition photo."

De la misma manera en la que se ha realizado en los casos de uso mencionadas anteriormente, como se puede corroborar en la la figura 7.11, la primera etapa de análisis de DENISE consiste en asignar una temática al texto que ha sido ingresado. En este caso, como en los anteriores, se ha detectado la temática: *informática*.

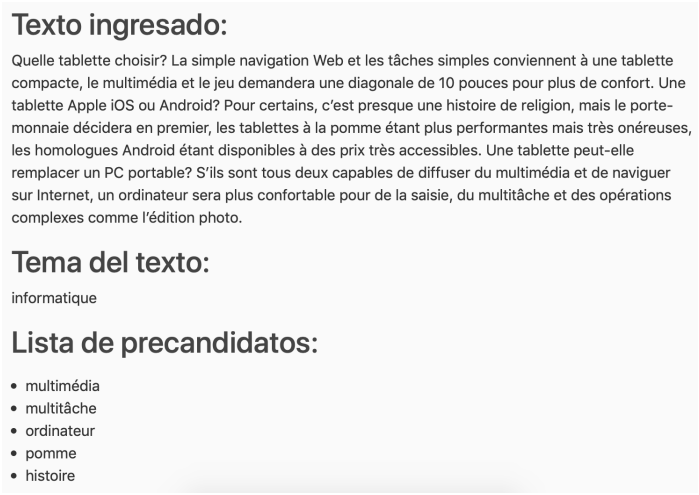


Figura 7.11 – Lista de precandidatos obtenida con DENISE en francés.

Después de realizar la clasificación de tema, el sistema procede a extraer precandidatos para obtener sus campos semánticos, que posteriormente serán empleados para realizar el proceso de desambiguación de tema. En este caso se han extraído los siguientes posibles candidatos (ver figura 7.12) a neologismo semántico: *histoire*, *multimédia*, *multitâche*, *ordinateur* y *pomme*[7]. Las unidades *ordinateur* y *multimêdia* han sido descartadas como candidatos ya que sus palabras más similares crean un campo semántico similar o perteneciente a la categoría *informática*.

[6] `https://guide-achat.lefigaro.fr/smartphones-tablettes-et-accessoires/tablettes/comparatif-quelle-est-la-meilleure-tablette-2018--5afb0d36-a608-11e8-b7e1-204ccac6f9c1/`

[7] Historia, multimedia, multitarea, ordenador y manzana en español.

Posibles candidatos a neologismo semántico:

Unidad	POS	Tema de embeddings	Estatus
multimédia	VERB	informatique	no candidato
multitâche	NOUN	otro	candidato
ordinateur	NOUN	informatique	no candidato
pomme	NOUN	otro	candidato
histoire	NOUN	otro	candidato

Figura 7.12 – Lista de posibles candidatos detectados.

Los posibles candidatos restantes (*multitâche*, *histoire* y *pomme*) fueron detectados como posibles candidatos ya que sus listados de palabras similares corresponden a una categoría distinta a la informática. En la figura 7.13 podemos comprobar que *histoire* no es una unidad realmente neológica, ya que sus similitudes corresponden al mismo significado empleado en el texto. En el caso de *multitâche*, podemos hablar de una instancia de falso positivo, ya que ha sido clasificado como no perteneciente a *informática* y al analizar las palabras más relacionadas con esta unidad, podemos comprobar que, en efecto, *multitâche* debería haber sido clasificado con la temática *informática*.

histoire:

Entrada	Similitud	Frecuencia
récit	0.5469	1182
histoires	0.5343	11131
narration	0.5332	2198
épopée	0.5264	2228
récits	0.5241	7426

(a) Histoire

pomme:

Entrada	Similitud	Frecuencia
pommes	0.8129	4780
tomate	0.7666	1408
betterave	0.7139	356
haricot	0.7012	900
carotte	0.7005	573

(b) Pomme

multitâche:

Entrada	Similitud	Frecuencia
nativement	0.8398	239
OpenGL	0.8303	371
plugin	0.8271	162
Silverlight	0.8244	84
Z80	0.824	188

(c) Multitâche

ordinateur:

Entrada	Similitud	Frecuencia
ordinateurs	0.7753	5416
microprocesseur	0.6652	944
processeur	0.6486	3898
calculateur	0.6434	686
utilisateur	0.6395	8067

(d) Ordinateur

Figura 7.13 – Listado de palabras similares por candidato a NS en francés.

Por último, *pomme*, ha sido clasificado correctamente como no perteneciente a la categoría informática y, al comprobar sus palabras más similares, podemos ver que las palabras más relacionadas con esta unidad dan cuenta del significado de *fruta*. En el texto, *pomme*, se usa de forma metonímica para hacer referencia a la marca Apple, cuyo nombre traducido al francés es *pomme* y su logotipo una manzana. Este uso se ve reforzado mediante la comparación “el homólogo de Android”, otra marca fabricante de tabletas

mencionada en el artículo. Mientras que el caso de *pomme* puede considerarse neológico, restaría analizar la frecuencia de uso para corroborar su estabilidad en la lengua francesa.

DENISE es una herramienta cuya finalidad es agilizar la detección de NS terminológicos. Mientras que la metodología manual requiere consultar todas las fuentes lexicográficas que se tengan a disposición para corroborar que una palabra ha sufrido un proceso de cambio de significado. Esta herramienta sustituye estas fuentes por un modelo de lengua que genera campos semánticos de la acepción más común dentro del modelo para llevar a cabo una desambiguación mediante el análisis de concordancia de tema.

Capítulo 8

CONCLUSIONES

En esta tesis hemos desarrollamos un estudio sobre la detección semiautomática de neologismos semánticos desde una perspectiva teórica y aplicada. Desde el punto de vista teórico, un estado del arte de las diferentes teorías sobre neologismos y su clasificación se ha realizado con la finalidad de comprender sus fundamentos y definir sus características y diferencias. Puesto que las unidades de interés se encuentran acotadas a neologismos semánticos terminológicos, también presentamos conceptos sobre la neología terminológica y las unidades terminológicas.

Desde el punto de vista aplicado, revisamos las propuestas contemporáneas en materia de detección y extracción automática. Encontramos cuatro propuestas metodológicas actuales, que sirvieron como guía para comparar los resultados que se pueden esperar de cada una y las posibles propuestas de evaluación. A partir de este análisis, diseñamos un primer acercamiento que, si bien no dio los resultados esperados, sirvió como punto de comparación y fue clave para descartar el uso de recursos de lingüísticos (fuentes de conocimiento lexicográfico y medidas de similitud).

Tras estos resultados iniciales, desarrollamos una segunda metodología que combina estrategias de aprendizaje automático supervisado y aprendizaje profundo, en conjunto con una implementación de TextRank, para extraer los precandidatos a neologismos semánticos. Por una parte, analizamos los modelos de clasificación de documentos comúnmente usados dentro del campo del aprendizaje automático y, por otra parte, comparamos tres modelos de representaciones distribuidas de palabras (*word embeddings*) para seleccionar los modelos que podrían dar los mejores resultados.

A partir de estos elementos, desarrollamos un algoritmo que emplea el análisis de la concordancia de temas como metodología para la detección de neologismos semánticos. Tras llevar a cabo la evaluación de los componentes de esta metodología, desarrollamos una implementación Web cuya finalidad es analizar un texto de entrada y presentar al usuario un listado de posibles candidatos a neologismo semántico.

8.1 Respuestas a los objetivos de la tesis

En el marco teórico hemos **analizado las características que definen y permiten identificar a los neologismos semánticos**. Como recursos específicos para su creación, podemos mencionar los siguientes mecanismos: figuras retóricas como la metáfora y la metonimia; nombres propios que pueden incluir la integración de marcas y nombres procedentes

de una lengua extranjera y, finalmente, la innovación terminológica científico–técnica.

También pudimos contrastar dos visiones contemporáneas sobre la clasificación de los neologismos y el lugar que ocupan los neologismos semánticos. Por una parte la tipología basada en matrices lexicogénicas de Sablayrolles (2000) concibe los neologismos semánticos como un tipo de neología de matriz interna. Por otra parte la tipología de Cabré (2011b) indica que los neologismos semánticos se ubican dentro de los procesos de formación, como un proceso de cambio de significado por resemantización que puede deberse a tres mecanismos: reducción, ampliación o cambio de significado.

Hemos constatado que las **aproximaciones actuales para la detección y extracción de neologismos semánticos (NS)** intentan dar respuesta a esta problemática desde diferentes enfoques. Por ejemplo, la clasificación y modelado de temas y la inducción de significado. No obstante, estos enfoques no proponen un flujo de trabajo concreto y definido que dé como resultado candidatos a NS. Las propuestas actuales son estrategias que pueden ser implementadas para este fin que, a pesar de que evalúan las estrategias que implementan, no establecen una línea base que pueda usarse como punto de comparación.

Así, tras la revisión del estado de la cuestión hemos observado que en materia de detección de candidatos a neologismos semánticos, uno de los enfoques más interesantes es el análisis de la concordancia de temas. Falk et al. (2014b) sugieren que este enfoque puede ser empleado para diseñar un sistema que sea capaz de detectar neologismos semánticos automáticamente, no obstante, no se implementa dentro de la plataforma Logoscope y tampoco se presenta una evaluación formal de esta metodología.

Nuestro sistema DENISE intenta cubrir este vacío proponiendo una arquitectura que se compone de diferentes piezas intercambiables que son, a nuestro juicio, los elementos básicos para llevar a cabo la detección de nuevos significados: una fuente de conocimiento que sirva para desambiguar los temas prototípicos de una palabra, una metodología de extracción de palabras y un modelo de clasificación de temas para analizar los textos ingresados. Con estos componentes, el sistema busca desambiguar los temas del texto que se analiza y los campos semánticos de los candidatos para detectar candidatos a neologismos semánticos terminológicos, en concreto relacionados con la informática.

Nuestro algoritmo combina estas piezas para comprobar la existencia de nuevos significados, compara los temas del texto que esta siendo analizado y el tema de la lista de palabras más comunes de cada precandidato que ha sido detectado. Esta arquitectura puede ser mejorada empleando metodologías de extracción de palabras claves más precisas. Sin embargo, DENISE —como *pipeline*— establece una linea base para comprobar la eficacia y precisión de versiones posteriores o de otras herramientas diseñadas para el mismo propósito.

Realizamos múltiples etapas de **evaluación del sistema**. Evaluamos cinco modelos de clasificación automática y tres modelos de representaciones vectoriales de palabras. La evaluación preliminar de metodologías de clasificación nos ha permitido corroborar que la regresión logística permite obtener una alta precisión y capacidad de generalización, comparable con metodologías como las máquinas de vectores de soporte (SVM) y el perceptrón multicapa, mientras que es un modelo simple en comparación con los antes mencionados. También comprobamos que los listados de palabras generados con *embeddings* pueden ser empleados para distinguir el significado y la temática prototípicos de una palabra.

Después de la presentación de este acercamiento metodológico y la evaluación de sus

componentes, **desarrollamos una herramienta** que, gracias a su implementación Web, pone a disposición de especialistas y el público general una plataforma cuya finalidad es facilitar la tarea de detección y extracción semiautomática de neologismos semánticos que provienen del campo de la informática. La implementación Web de DENISE genera un reporte de cada una de las etapas de procesamiento y presenta un listado con precandidatos y posibles candidatos, así como las temáticas de sus acepciones más comunes para que usuario seleccione un candidato válido, en el caso de que exista.

8.2 Limitaciones

La hipótesis de trabajo de DENISE se basa en la detección de temas desde una perspectiva de clasificación de documentos. Sin embargo, la selección de precandidatos depende de la extracción de palabras relevantes en el discurso y de la desambiguación de temas. Para llevar a cabo el proceso de extracción de palabras, utilizamos una implementación de TextRank con filtros de etiquetas gramaticales.

La decisión de implementar TextRank fue tomada, en parte, por una razón operativa: TextRank es una metodología de extracción de palabras clave extensamente implementada por su simplicidad y versatilidad. También por una razón metodológica pues el proceso de extracción basado en un sistema de *voto* por relevancia de TextRank, parecía una alternativa viable para extraer precandidatos.

Durante la evaluación de esta metodología (ver tabla 8.1), observamos que cuando se trata de extracción de palabras claves a partir de contextos cortos, la precisión de este algoritmo disminuye. Por lo tanto, una mejora clara es la implementación de una metodología de extracción de palabras que permita obtener mejores candidatos sin importar la longitud del contexto. Puesto que el propósito de nuestro enfoque era comprobar que la discordancia de temas es un factor clave para la detección de neologismos semánticos, no se llevó a cabo una evaluación de algoritmos de extracción de palabras.

	Casos	Recuperado	Porcentaje
Formas CA	120	42	35.00 %
Lemas CA	120	36	30.00 %
Formas ES	194	70	36.08 %
Lemas ES	194	69	35.57 %

Tabla 8.1 – Resumen de palabras claves obtenidas con TextRank por formas y lemas en catalán y español.

Antes de desarrollar una segunda versión de DENISE, sería de interés realizar una comparación de algoritmos de extracción de palabras clave a partir de contextos cortos, ya que dichas metodologías generalmente son evaluadas con textos más extensos, por ejemplo: resúmenes, reseñas o artículos académicos. Creemos que la extracción de palabras claves es un componente clave puesto que, además de generar un listado de precandidatos, es una etapa de filtrado de palabras que descarta aquellas que no son de interés o que tienen menor probabilidad de ser candidatos a neologismo semántico.

En cuanto a los modelos neuronales de lengua, a pesar de que el uso de *embeddings* en el campo del procesamiento del lenguaje natural se ha convertido en una práctica usual,

las representaciones de palabras similares varían en función del modelo y arquitectura empleada. La tabla 8.2 muestra tres casos de listas de palabras similares que fueron extraídas durante el proceso de revisión manual. Estas similitudes no son útiles para nuestro sistema, ya que no aportan la información suficiente para poder ser clasificados correctamente.

En el primer caso, denominado *no informativo*, podemos señalar que las palabras más similares a *dominio* son variaciones de la misma palabra. El segundo contempla representaciones *ambiguas* y está constituido por similitudes que muestran dos campos semánticos distintos en un mismo listado: la primera mitad del listado de similitudes de *nube* corresponde a un significado propio de la informática, mientras que la segunda mitad al significado meteorológico. En tercer término se encuentran las representaciones de *L2 en L1*, podemos ver que *palm*, en nuestro caso un dispositivo portátil, ha generado similitudes con otras palabras del inglés (L2) que son usadas dentro del texto en español (L1).

No informativo	Ambiguo	L2 en L1
Dominio	Nube	Palm
Dominio	cloudcomputing	plum
dominios	OwnCloud	frog
dominio.El	SoftLayer	wood
dominio.	ownCloud	leaved
eldominio	IaaS	lily
sub-dominio	nubecitas	ferns
dominio.En	neblina	oak
dominio.-	virtualizada	apple
dedominio	NUBE	maple
domino	SaaS	gum
subdominio	hiperconvergencia	found_in
dominiode	niebla	native
dominio.La	clouds	leaf
dominio.com	Wordle	fruit
dominio-	vaporosa	jelly

Tabla 8.2 – Tipos de listados de similitud no útiles para desambiguación de tema.

Finalmente, el modelo de clasificación de documentos basado en regresión logística dio los mejores resultados con nuestro corpus de entrenamiento. Sin embargo, en la actualidad existen metodologías de clasificación de documentos basadas en redes neuronales; por ejemplo, modelos de clasificación automática mediante redes neuronales convolucionales (CNN), redes neuronales recurrentes (RNN) o redes de atención jerárquica (HAN) que podrían ser implementados.

A pesar de que los los modelos neuronales antes mencionados tienen mayor complejidad, su implementación y entrenamiento pueden ser comparables con el modelo de regresión logística. En conjunto con la evaluación de estas tres metodologías, también se tendría que considerar la constitución de conjuntos de datos más amplios y con más temáticas, de forma que el algoritmo de detección no se encuentre limitado a la informática.

En resumen, las tres mejoras claves que podrían resultar en una plataforma más sólida serían:

- Seleccionar una metodología alternativa a TextRank para la extracción de precandidatos.
- Implementar un modelo de clasificación de documentos con una arquitectura neuronal.
- La adición de fuentes de conocimiento externo que complementen a los modelos de *embeddings*.

8.3 Líneas de investigación futuras

A partir del análisis de las características intrínsecas de los neologismos semánticos, podemos mencionar dos líneas de investigación futuras:

- Análisis computacional de la metáfora como recurso para la creación de NS.
- Creación de un modelo polisémico de representaciones vectoriales de palabras.

La primera línea se encuentra relacionada con uno de los mecanismos de creación de neologismos semánticos: la metáfora. En conjunto con la desambiguación de temas, la detección de usos metafóricos de palabras puede funcionar como un filtro adicional durante la extracción de precandidatos a neologismo semántico. Bajo este nuevo supuesto, no basta que una palabra sea relevante dentro del texto que se analiza y que exista discordancia de temáticas del significado novedoso y convencional, sino que el contexto analizado debe contener un uso metafórico.

En la actualidad, las metáforas se analizan desde distintas perspectivas computacionales (Shutova, 2010); ya sea partiendo del concepto de metáfora conceptual (Veale et al., 2016) o desde otras perspectivas puramente computacionales. Estas últimas emplean métodos como el aprendizaje profundo (Tsvetkov et al., 2014; Tanasescu et al., 2018; Rosen, 2018) para clasificar y detectar usos metafóricos de palabras y textos que pueden contener enunciados con expresiones metafóricas. Recientemente han surgido avances en la detección de metáforas *nombre + adjetivo* desde el aprendizaje profundo (Bizzoni et al., 2017). No obstante no se ha implementado el análisis de este tipo de figuras retóricas para la detección de neologismos.

Como segunda línea de investigación, consideramos el desarrollo de modelos de *embeddings* polisémicos. Este tipo de acercamiento se encuentra en auge (Camacho-Collados y Pilehvar, 2018). Existen enfoques como el desarrollo de técnicas posteriores al entrenamiento para reorganizar las representaciones vectoriales que han sido generadas (Sun et al., 2017). Añadir información semántica de fuentes externas (Pilehvar y Collier, 2016; Mancini et al., 2017; Pilehvar et al., 2017), estrategias de desambiguación e inducción de significado durante el mismo proceso de entrenamiento (Liu et al., 2015; Kekeç et al., 2018; Arora et al., 2018) y combinaciones lineares de las representaciones vectoriales (Hu et al., 2016).

Mientras que el modelo Sense2Vec que fue implementado para el desarrollo de nuestro sistema genera múltiples representaciones al utilizar un corpus de entrenamiento etiquetado gramaticalmente, existen algunos casos conflictivos:

- Algunas de las representaciones generadas no corresponden a la etiqueta gramatical asignada. Por lo tanto, cuando se realizan consultas indicando manualmente la etiqueta gramatical que corresponde a una palabra, este modelo no muestra la información adecuada.

- No es un modelo que genere verdaderas representaciones polisémicas ya que no lleva a cabo un proceso de desambiguación semántica. En cambio, genera una representación por cada una de las etiquetas gramaticales signadas a una palabra. En este escenario, la representación vectorial de una etiqueta gramatical puede tener similitudes que agrupan dos o más conceptos semánticamente distintos, no resolviendo el problema de la ambigüedad semántica.

El diseño de un modelo de *word embeddings* que tenga la capacidad de generar representaciones polisémicas y que al mismo tiempo se encuentre etiquetado gramaticalmente, podría ser una aportación importante. Dicho modelo permitiría realizar con mayor precisión tareas como la traducción automática, generación de texto, resolución de referencias, extracción de terminología y también como base para el entrenamiento de modelos de clasificación automática.

Capítulo 9

CONCLUSIONS

Dans cette thèse nous réalisons une étude sur la détection semi-automatique des néologismes sémantiques à partir d'une perspective théorique et appliquée. Du point de vue théorique, un état de l'art des différentes théories sur les néologismes et leur classification a été réalisé afin de comprendre leurs fondations et établir leurs caractéristiques et leurs différences. Puisque les unités d'intérêt sont limitées aux néologismes sémantiques terminologiques, nous présentons également des concepts de néologie terminologique et d'unités terminologiques.

Du point de vue appliqué, nous examinons les propositions actuelles concernant la détection et l'extraction automatique de néologismes. Quatre de ces propositions méthodologiques principales ont servi de guide pour déterminer le type de méthodologies utilisées, leurs résultats attendus et des propositions d'évaluation possibles. À partir de cette étude, nous avons conçu une première approche qui, bien qu'elle n'ait pas donné les résultats escomptés, a servi de point de comparaison et a été essentielle pour s'affranchir du besoin de ressources linguistiques (des connaissances lexicographiques et des mesures de similarité).

Suite à ces premiers résultats, nous avons développé une autre méthodologie combinant des stratégies d'apprentissage automatique supervisé et profond, avec une implémentation de TextRank, afin d'extraire les candidats au néologisme sémantique. D'une part, nous avons analysé les modèles de classification des documents couramment utilisés dans le domaine de l'apprentissage automatique et, d'autre part, nous avons comparé trois modèles de plongement de mots (*word embedding*) pour sélectionner les modèles le plus performants.

A partir de ces éléments, nous avons développé un algorithme qui utilise l'analyse de concordances de thèmes comme méthodologie pour la détection de néologismes sémantiques. Après une évaluation des composants de cette méthodologie, nous avons mis au point une implémentation Web destinée à analyser des documents textuelles et présenter à l'utilisateur une liste de candidats possibles au néologisme sémantique.

9.1 Réponses aux objectifs de la thèse

Dans la partie théorique, nous avons **analysé les caractéristiques qui définissent et permettent d'identifier les néologismes sémantiques**. Parmi les ressources spécifiques utilisées pour les identifier, on peut citer les mécanismes suivants : les figures rhétoriques,

telles que la métaphore et la métonymie ; les noms propres pouvant inclure l'intégration de marques et de noms en langue étrangère et enfin, l'innovation terminologique scientifique–technique.

Nous avons également pu confronter deux conceptions contemporaines de la classification des néologismes et y positionner les néologismes sémantiques. D'une part, la typologie basée sur les matrices lexicogènes de Sablayrolles (2000) qui conçoit les néologismes sémantiques comme un type de néologie de matrice interne. D'autre part, la typologie de Cabré (2011b) qui indique que les néologismes sémantiques se situent dans les processus de formation, en tant que processus de changement de sens par resémantisation pouvant être dû à trois mécanismes : la réduction, l'extension ou le changement de sens.

Nous avons constaté que **les approches actuelles de détection et d'extraction de néologismes sémantiques (NS)** essayaient de répondre à cette problématique par différentes approches. Par exemple, par la modélisation et classification thématiques et l'induction de sens. Cependant, ces approches ne proposent pas un processus concret et défini qui montre comme résultat une liste de candidats au néologisme sémantique. Ces méthodes sont des différentes stratégies qui, bien qu'elles sont évaluées, elles n'offrent pas un référentiel (baseline) qui permettrait de les comparer.

Ainsi, après avoir examiné l'état de l'art, nous avons constaté qu'en pour la détection de candidats au néologisme sémantique, l'une des approches le plus intéressante est celui de ll'analyse de concordance thématique. Falk et al. (2014b) suggèrent que cette approche pourrait être utilisée pour concevoir un système capable de détecter automatiquement les néologismes sémantiques, cependant cela n'a pas été implanté dans leur système Logoscope, et aucune évaluation formelle de cette méthodologie n'a été présentée.

Notre système DENISE cherche à combler ce manque en proposant une architecture composée de modules interchangeables qui sont, à notre avis, les éléments de base pour la détection de nouveaux sens : une ressource de connaissances qui aide à désambiguïser les thématiques prototypiques d'un mot, une méthodologie d'extraction de mots et un modèle de classification thématique pour l'analyse des textes. Avec ces composants, le système cherche à désambiguïser les thématiques du document analysé et les champs sémantiques des candidats afin de détecter les candidats au néologisme sémantique terminologiques, spécifiquement ceux liés aux technologies de l'information.

Notre algorithme combiné ces éléments pour vérifier l'existence de nouvelles significations, compare les thématiques du texte analysé et la thématique de la liste des mots les plus courants de chaque candidat détecté. Cette architecture peut être améliorée à l'aide de méthodes d'extraction de mots clés plus précises. Cependant, DENISE en tant que chaîne de traitement (ou *pipeline*) établit un référentiel (baseline) pour évaluer la performance et la précision des nouvelles versions ou d'autres outils conçus dans le même but.

Nous avons réalisé **multiples étapes d'évaluation de notre système** et nous avons valorisé plusieurs modèles de classification automatique et de représentation vectorielle de mots. L'évaluation préliminaire des méthodes de classification nous a permis de confirmer que la régression logistique permet d'obtenir une précision et une capacité de généralisation élevées, comparables à celles obtenues par les méthodes telles que les machines à vecteurs support (SVM) et les perceptrons multicouches, alors que notre modèle est bien plus simple. Nous avons également constaté que les listes de mots générées avec les plongement des mots peuvent être utilisées pour distinguer signification et thématique

prototypique d'un mot.

Ainsi, outre la présentation de cette approche méthodologique et l'évaluation de ses composants, nous avons **développé un outil** qui, grâce à son implémentation Web, met à la disposition des spécialistes et du grand public une plate-forme destinée à faciliter la tâche de détection et d'extraction semi-automatique de néologismes sémantiques issus du domaine de l'informatique. L'implémentation Web de DENISE génère un rapport avec les informations de chacune des étapes du traitement et présente une liste des précandidats et candidats possibles, ainsi que les thématiques de leurs significations les plus courantes afin que l'utilisateur sélectionne un candidat valide, le cas échéant.

9.2 Limitations

L'hypothèse de travail de DENISE est basée sur la détection de thématiques dans une perspective de classification de documents. Cependant, la sélection de précandidats dépend de l'extraction de mots pertinents dans le discours et de la désambiguïsation des thématiques. Pour mener à bien le processus d'extraction de mots, nous avons utilisé une implémentation TextRank avec des filtres d'étiquetage grammatical.

La décision d'implémenter TextRank s'explique par deux raisons : une raison opérationnelle et une autre méthodologique. La raison opérationnelle fait référence au fait que TextRank est une méthode d'extraction de mots-clés largement mise en œuvre pour sa simplicité et sa polyvalence. La raison méthodologique par le fait que le processus d'extraction basé sur un système de graphes et de vote par pertinence est une alternative viable pour extraire les précandidats.

Lors de l'évaluation de cette méthodologie (voir Table 9.1), nous avons observé que lorsqu'il était question d'extraire des mots-clés à partir de contextes courts, la précision diminuait. Ainsi une amélioration évidente a été la mise en œuvre d'une méthodologie d'extraction de mots permettant d'obtenir de meilleurs candidats quelle que soit la taille du contexte. Étant donné que le objectif de notre approche est constater que la discordance des thématiques est un facteur clé pour la détection des néologismes sémantiques, aucune évaluation des algorithmes d'extraction de mots n'a été réalisée.

	Cases	Récupéré	Pourcentage
Formes CA	120	42	35.00%
Lemmes CA	120	36	30.00%
Formes ES	194	70	36.08%
Lemmes ES	194	69	35.57%

TABLE 9.1 – Résumé des mots-clés obtenus avec TextRank par formes et lemmes en catalan et en espagnol.

Avant de développer une deuxième version de DENISE, il a été intéressant de réaliser une comparaison des algorithmes d'extraction de mots-clés à partir des contextes courts, car ces méthodologies sont généralement évaluées à l'aide de textes plus longs. Par exemple des résumés, des notices ou des articles académiques. Nous pensons que l'extraction de mots-clés est un élément essentiel car, en plus de générer une liste de pré-

candidats, cela constitue une étape de filtrage des mots qui élimine ceux qui ne présentent pas d'intérêt ou qui sont moins susceptibles d'être candidats au néologisme sémantique.

En ce qui concerne les modèles neuronaux, bien que l'utilisation de plongements de mots sont devenus une pratique habituelle dans le domaine du Traitement Automatique de Langues, la représentation des mots similaires est dépendante du modèle et de l'architecture utilisés. La table 9.2 présente trois cas de listes de mots similaires extraits au cours du processus de révision manuelle. Ces similitudes ne sont pas utiles pour notre système, car elles ne fournissent pas l'information requise pour que ces mots soient correctement classés.

Au premier cas, appelé *non informatif*, il est possible de remarquer que les mots les plus similaires au mot « dominio » (domaine) sont des variantes du même mot. Le deuxième considère les plongements de *mots ambigus* et il est constitué des similitudes qui montrent deux champs sémantiques différents dans la même liste : dans la première moitié de la liste, les similitudes du mot « nube » (nuage) correspondente à une signification propre à l'informatique, en tant que la deuxième moitié correspond à la signification météorologique. Dans le troisième cas, nous trouvons les représentations de *L2 dans L1* et nous pouvons voir que le mot « palm » , dans notre cas s'agissant d'un téléphone portable, a généré des similitudes avec d'autres mots en anglais (L2) qui sont utilisés en espagnol (L1).

Non informatif	Ambigües	L2 dans L1
Dominio	Nube	Palm
Dominio	cloudcomputing	plum
dominios	OwnCloud	frog
dominio.El	SoftLayer	wood
dominio.	ownCloud	leaved
eldominio	IaaS	lily
sub-dominio	nubecitas	ferns
dominio.En	neblina	oak
dominio.-	virtualizada	apple
dedominio	NUBE	maple
domino	SaaS	gum
subdominio	hiperconvergencia	found_in
dominiode	niebla	native
dominio.La	clouds	leaf
dominio.com	Wordle	fruit
dominio-	vaporosa	jelly

TABLE 9.2 – Types de listes de similarité non utiles pour la désambiguïsation des thèmes.

Finalement, le modèle de classification de documents basé sur la régression logistique a obtenu les meilleurs résultats avec notre corpus d'apprentissage. Cependant, il existe actuellement des méthodologies de classification de documents basées sur des réseaux de neurones ; par exemple, les modèles de classification automatique par réseaux de neurones convolutionnels (CNN), par réseaux de neurones récurrents (RNN) ou par réseaux d'attention hiérarchiques (HAN) qui peuvent être testées.

Bien que les modèles neuronaux mentionnés soient plus complexes, leur mise en œuvre et l'étape d'apprentissage peuvent être comparables au modèle de régression logistique. Parallèlement à l'évaluation de ces trois méthodologies, il convient également d'envisager la construction des ensembles de données plus vastes et des thématiques plus nombreuses afin que l'algorithme de détection puisse être utilisé dans des domaines autres que l'informatique.

En résumé, les trois améliorations clefs qui pourraient donner lieu à une plateforme plus solide sont les suivantes :

- Choisir une méthodologie alternative à TextRank pour l'extraction des précandidats.
- La lise en œuvre un modèle de classification de documents avec une architecture neuronale.
- Ajouter des ressources de connaissances externes pour compléter les modèles par plongements des mots.

9.3 Lignes de recherche futures

Sur la base de l'analyse des caractéristiques intrinsèques des néologismes sémantiques, nous pouvons citer deux axes de recherche futurs :

- L'analyse de la métaphore en tant que ressource pour la création de NS.
- Le développement d'un modèle polysémique de représentations vectorielles de mots.

Le premier axe est lié à l'un des mécanismes de création de néologismes sémantiques : la métaphore. Au même temps que la désambiguïsation des thématiques, la détection d'utilisation de mots métaphoriques peut servir de filtre supplémentaire lors de l'extraction de mots-clés. Sous cette nouvelle hypothèse, il ne suffit pas qu'un mot soit pertinent dans le texte analysé et qu'il existe un désaccord thématique entre les significations nouvelle et conventionnelle, mais également que le contexte analysé ait un usage métaphorique.

À l'heure actuelle, les métaphores sont analysées selon différentes perspectives informatiques (Shutova, 2010) ; à partir du concept de métaphore conceptuelle (Veale et al., 2016) ou à partir d'autres perspectives purement informatiques. Ces cernières utilisent des méthodes telles que l'apprentissage profonde (Tsvetkov et al., 2014; Tanasescu et al., 2018; Rosen, 2018)pour classifier et détecter les utilisations métaphoriques de mots et de textes pouvant contenir des phrases comportant des expressions métaphoriques. Récemment, des progrès ont été réalisés dans la détection de métaphores « nom + adjectif » avec l'apprentissage profond (Bizzoni et al., 2017). Cependant, l'analyse de ce type de figures rhétoriques pour la détection de néologismes n'a pas été implémentée.

Quant a la deuxième ligne de recherche, nous considérons le développement de modèles de plongements de mots polysémiques. Ce type d'approches sont en pleine croissance(Camacho-Collados y Pilehvar, 2018). Elles sont notamment basés sur le développement de techniques d'apprentissage visant à réorganiser les représentations vectorielles

générées (Sun et al., 2017). Elles ajoutent des informations sémantiques issues de ressources externes (Pilehvar y Collier, 2016; Mancini et al., 2017; Pilehvar et al., 2017), des stratégies de désambiguïsation et d'induction de sens pendant le processus d'apprentissage (Liu et al., 2015; Kekeç et al., 2018; Arora et al., 2018) ainsi que des combinaisons linéaires des représentations vectorielles ((Hu et al., 2016). Bien que le modèle Sense2Vec génère plusieurs représentations en utilisant un corpus d'apprentissage étiqueté grammaticalement, il existe quelques cas contradictoires :

- Certaines représentations générées ne correspondent pas à l'étiquette grammaticale affectée. Donc, lors de requêtes indiquant manuellement l'étiquetage grammaticale correspondant à un mot, ce modèle n'affiche pas les informations appropriées.
- Ce modèle ne génère pas de véritables représentations polysémiques, car il ne réalise aucun processus de désambiguïsation sémantique. Au lieu de cela, il génère une représentation pour chacune des étiquettes grammaticales attribuée à un mot. Dans ce scénario, la représentation vectorielle d'un étiquetage grammatical peut avoir des similitudes qui regroupent deux ou plusieurs concepts sémantiquement différents, mais le problème d'ambiguïté sémantique reste encore.

La conception d'un modèle de plongements de mots capable de générer des représentations polysémiques et en même temps de trouver une étiquette grammaticale, pourrait constituer une contribution importante. Cela permettrait la réalisation plus précisément des tâches telles que la traduction automatique, la génération de texte, la résolution de références, l'extraction de terminologie et aussi servir de base pour l'apprentissage de modèles de classification automatique.

Bibliografía

Adelstein, A. (1996). Banalización de términos con formantes de origen grecolatino. *Simposio Iberoanericano de Terminología RITerm*, 5.

Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv:1803.08375 [cs, stat]*.

Al-Rfou, R., Perozzi, B., y Skiena, S. (2013). Polyglot: Distributed word representations for multilingual nlp. En *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pp. 183–192, Sofia, Bulgaria. Association for Computational Linguistics.

Aletras, N., Baldwin, T., Lau, J. H., y Stevenson, M. (2014). Representing topics labels for exploring digital libraries. En *Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '14, pp. 239–248, Piscataway, NJ, USA. IEEE Press.

Allen, J. (1995). *Natural Language Understanding (2Nd Ed.)*. Benjamin-Cummings Publishing Co., Inc., Redwood City, CA, USA.

Arora, S., Li, Y., Liang, Y., Ma, T., y Risteski, A. (2018). Linear Algebraic Structure of Word Senses, with Applications to Polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495. http://dx.doi.org/10.1162/tacl_a_00034.

Bach, C. y Cabré, M. T. (2004). El corpus tècnic del IULA: corpus textual especializado plurilingüe. *Panace@: Boletín de Medicina y Traducción*, 5(16):173–176.

Baeza-Yates, R. y Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. ACM Press, New York.

Banerjee, M., Capozzoli, M., McSweeney, L., y Sinha, D. (1999). Beyond kappa: A review of interrater agreement measures. *Canadian Journal of Statistics*, 27(1):3–23. http://dx.doi.org/10.2307/3315487.

Barrios, F., López, F., Argerich, L., y Wachenchauzer, R. (2016). Variations of the Similarity Function of TextRank for Automated Summarization. *CoRR*, abs/1602.0.

Barzilai, J. y Borwein, J. M. (1988). Two-Point Step Size Gradient Methods. *IMA Journal of Numerical Analysis*, 8(1):141–148. http://dx.doi.org/10.1093/imanum/8.1.141.

Bastuji, J. (1974). Aspects de la néologie sémantique. http://dx.doi.org/10.3406/lgge.1974.2270.

Bengio, Y. (2009). Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, 2(1):1–127.

Bengio, Y. (2012). Deep Learning of Representations for Unsupervised and Transfer Learning. En *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pp. 17–36.

Bengio, Y., Ducharme, R., Vincent, P., y Jauvin, C. (2003). A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, 3:19.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Information science and statistics. Springer, New York.

Bizzoni, Y., Chatzikyriakidis, S., y Ghanimifard, M. (2017). "Deep"Learning : Detecting Metaphoricity in Adjective-Noun Pairs. En *Proceedings of the Workshop on Stylistic Variation*, pp. 43–52, Copenhagen, Dinamarca. Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/W17-4906.

Bojanowski, P., Grave, E., Joulin, A., y Mikolov, T. (2016). Enriching Word Vectors with Subword Information. *arXiv:1607.04606 [cs]*.

Boser, B. E., Guyon, I., y Vapnik, V. (1992). A training algorithm for optimal margin classifiers. En *Proceedings of the fifth Annual ACM Conference on Computational Learning Theory, COLT 1992, Pittsburgh, PA, USA, July 27-29, 1992.*, pp. 144–152. http://dx.doi.org/10.1145/130385.130401.

Bottou, L., Curtis, F. E., y Nocedal, J. (2016). Optimization Methods for Large-Scale Machine Learning. *arXiv:1606.04838 [cs, math, stat]*.

Boulanger, J.-C. (1991). Une lecture socioculturelle de la terminologie. *Cahiers de Linguistique Sociale*, 18:13–30.

Boussidan, A. y Ploux, S. (2011). Using topic salience and connotational drifts to detect candidates to semantic change. En *Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)*.

Breiman, L. (1996). Bagging predictors. En *Machine Learning*, pp. 123–140, Hingham, MA, USA. Kluwer Academic Publishers.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32. http://dx.doi.org/10.1023/A:1010933404324.

Cabré, M. T. (1992). *La Terminologia: la teoria, els mètodes, les aplicacions*. Empúries, Barcelona.

Cabré, M. T. (1999). *La terminología: representación y comunicación: elementos para una teoría de base comunicativa y otros artículos*. Institut Universitari de Lingüística Aplicada, Universitat Pompeu Fabra, Barcelona.

Cabré, M. T. (2004). La importància de la neologia per al desenvolupament sostenible de la llengua catalana. En Observatori de Neologia, editor, *Llengua catalana i neologia*, pp. 17–45. Meteora, Barcelona.

Cabré, M. T. (2009). La classificació dels neologismes: una tasca complexa. En Cabré, M. T. y Estopà, R., editores, *Les paraules noves: criteris per detectar i mesurar els neologismes*, pp. 11–37. Eumo Editorial, Universitat Pompeu Fabra, Vic, Barcelona.

Cabré, M. T. (2011a). El principio de la poliedricidad: La articulación de lo discursivo, lo congnitivo y lo lingüístico en terminología. *Organon*, 25(50). http://dx.doi.org/10.22456/2238-8915.28343.

Cabré, M. T. (2011b). La neología y los neologismos: reflexiones teóricas y cuestiones aplicadas. En Vázques Laslop, M. E., Zimmermann, K., y Segovia, F., editores, *De la lengua por sólo la extrañeza: estudios de lexicología, norma lingüística, historia y literatura en homenaje a Luis Fernando Lara*, volumen 1. Colegio de México, México.

Cabré, M. T., Bach, C., y Vivaldi, J. (2006). 10 anys del Corpus de l'IULA. En Fabra, U. P., editor, *Papers del IULA, Sèrie Informes*, volumen 44. Institut Universitari de Lingüística Aplicada, Barcelona.

Cabré, M. T., Domènech, O., Estopà, R., Freixa, J., y Solé, E. (2004). La lexicografia i la detecció automatitzada de neologia lèxica. En Battaner, P. y DeCesaris, J., editores, *De Lexicografía*, pp. 287–294, Barcelona. Institut Universitari de Lingüística Aplicada.

Cabré, M. T., Domènech Bagaria, O., y Estopà, R. (2018). *La terminologia avui: termes, textos i aplicacions*. Editorial UOC, Barcelona.

Cabré, M. T. y Estopà, R. (2004a). Metodología del trabajo en neología: criterios, materiales y procesos. *Papers de l'IULA*.

Cabré, M. T. y Estopà, R. (2004b). Metodologia del treball en neologia: criteris, materials i processos. *Papers de l'IULA*.

Cabré, M. T. y Estopà, R. (2005). Unidades de conocimiento especializado: caracterización y tipología. En Cabré, M. T. y Carme, B., editores, *Coneixement, llenguatge i discurs especialitzat*, pp. 69–93. Institut Universitari de Lingüística Aplicada. Universitat Pompeu Fabra, Barcelona.

Cabré, M. T., Estopà, R., y Vivaldi, J. (2001). Automatic Term Detection: A Review of Current Systems. En Bourigault, D., Jacquemin, C., y LHomme, M., editores, *Recent Advances in Computational Terminology*, pp. 30–32. John Benjamins.

Camacho-Collados, J. y Pilehvar, M. T. (2018). From Word To Sense Embeddings: A Survey on Vector Representations of Meaning. *Journal of Artificial Intelligence Research*, 63:743–788. http://dx.doi.org/10.1613/jair.1.11259.

Caudill, M. (1987). Neural networks primer, part i. *AI Expert*, 2(12):46–52.

Caudill, M. y Butler, C. (1992). *Understanding Neural Networks; Computer Explorations*. MIT Press, Cambridge, MA, USA.

Chung, K. L. (1954). On a Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 25(3):463–483. http://dx.doi.org/10.1214/aoms/1177728716.

Church, K. W. y Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29. http://dx.doi.org/10.3115/981623.981633.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46. http://dx.doi.org/10.1177/001316446002000104.

Collier, A. (1998). Identifying diachronic change in semantic relations. En Renouf, A., editor, *Explorations in Corpus Lingüistics*. Rodopi, Amsterdam.

Collobert, R. y Weston, J. (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. *Proceedings of the 25th International Conference on Machine Learning*, p. 8. http://dx.doi.org/10.1145/1390156.1390177.

Conneau, A., Schwenk, H., Barrault, L., y Lecun, Y. (2017). Very Deep Convolutional Networks for Text Classification. *arXiv:1606.01781 [cs]*.

Cook, P. y Hirst, G. (2011). Automatic identification of words with novel but infrequent senses. En *Proceedings of the 25th Pacific Asia Conference on Language, Information and Computation*.

Cook, P. y Stevenson, S. (2010). Automatically identifying changes in the semantic orientation of words. En *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Languages Resources Association (ELRA).

Corbeil, J.-C. (1971). Aspects du problème néologique. *La Banque des Mots*, 2.

Cox, D. R. (1958). The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, 20(2):215–242.

Deloffre, F. (1985). Sur le vocabulaire de Rousseau, Rêveries du promeneur solitaire (Promenades V-X). *L Information Grammaticale*, 25(1):23–27. http://dx.doi.org/10.3406/igram.1985.2198.

Deng, L. y Liu, Y. (2018). *Deep Learning in Natural Language Processing*. Springer Publishing Company, Incorporated, 1st edición.

Domènech, O. (2008). Metodología de trabajo del observatorio de neología del iula. En Almela, R. y Montoro, E. T., editores, *Neologismo y morfología*, pp. 11–37, Murcia. Editum.

Dubin, D. (2004). The Most Influential Paper Gerard Salton Never Wrote. *Library Trends*, 52(4):748–764.

Duchi, J., Hazan, E., y Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *J. Mach. Learn. Res.*, 12:2121–2159.

Duda, O., R., Hart, E., P., y Stork, G., D. (2001). *Pattern Classification*. Wiley, second edition edición.

Estopà, R. (2009). Neologismes i filtres de neologicitat: aspectes metodògics. En Cabré, M. T. y Estopà, R., editores, *Les paraules noves: criteris per detectar i mesurar els neologismes*, pp. 39–48. Eumo Editorial, Universitat Pompeu Fabra, Vic, Barcelona.

Faber, P. (2012). *A Cognitive Linguistics View of Terminology and Specialized Language*. De Gruyter Mouton, Berlín; Boston. http://dx.doi.org/10.1515/9783110277203.

Faber, P. (2015). Frames as a framework for terminology. En Kockaert, H. J. y Steurs, F., editores, *Handbook of Terminology*, volumen 1, pp. 14–33. John Benjamins Publishing Company, Amsterdam. http://dx.doi.org/10.1075/hot.1.02fra1.

Falk, I., Bernhard, D., y Gérard, C. (2014a). De la quenelle culinaire à la quenelle politique: identification de changements sémantiques à l'aide des Topic Models. En *21ème conférence sur le Traitement Automatique des Langues Naturelles*, Marseille, Francia.

Falk, I., Bernhard, D., y Gérard, C. (2014b). From Non Word to New Word: Automatically Identifying Neologisms in French Newspapers. En *LREC - The 9th edition of the Language Resources and Evaluation Conference*, Proceedings of the International Conference on Language Resources and Evaluation, Reykjavik, Iceland.

Falk, I., Bernhard, D., Gérard, C., y Potier-Ferry, R. (2014c). Étiquetage morpho-syntaxique pour des mots nouveaux. En *21ème conférence sur le Traitement Automatique des Langues Naturelles*, Marseille, Francia.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., y Lin, C.-J. (2008). LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874.

Faulstich, E. (1995). *Base metodológica para pesquisa em socioterminologia: Termo e variaçao*. Universidade de Brasília. Departamento de Linguística, Línguas Clássicas e Vernácula, Brasilia.

Feliu, J., Martínez, L., Salazar, H. R., y Tadeo, B. (2009). Els neologismes formats per resemantització. En Cabré, M. T. y Estopà, R., editores, *Les paraules noves criteris per detectar i mesurar els neologismes*, pp. 89–110. Eumo editorial.

Firth, J. (1957). *Papers in Linguistics 1934-1951*. Oxford University Press University Press, Londres.

Fischer, M. M. (1998). Computational neural networks: A new paradigm for spatial analysis. *Environment and Planning A: Economy and Space*, 30(10):1873–1891. http://dx.doi.org/10.1068/a301873.

Fleiss, J. L., Cohen, J., y Everitt, B. S. (1969). Large sample standard errors of kappa and weighted kappa. *Psychological Bulletin*, 72(5):323–327. http://dx.doi.org/10.1037/h0028106.

Fletcher, R. (1987). *Practical Methods of Optimization; (2Nd Ed.)*. Wiley-Interscience, New York, NY, USA.

Freixa, J. (2009). NEOBANC: los observatorios de neología suman esfuerzos. *Debate Terminológico*, 5:120–125.

Gaudin, F. (1993). *Socioterminologie: des problèmes sémantiques aux pratiques institutionelles*. Número 182 en Publications de l'Université de Rouen. Publ. de l'Univ. de Rouen, Rouen.

Gérard, C., Falk, I., y Bernhard, D. (2014). Traitement automatisé de la néologie: pourquoi et comment intégrer l'analyse thématique? En *Actes du 4e Congrès Mondial de Linguistique Française (CMLF 2014)*, volumen 8 de *SHS Web of Conferences.*, pp. 2627 – 2646, Berlín, Alemania. http://dx.doi.org/10.1051/shsconf/20140801208.

Giardina, C. (1992). La Création Lexicale Dans L'écume des Jours de Boris Vian. *La Banque des Mots*, 1(44):63–83.

Glorot, X. y Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 9:249–256.

Glorot, X., Bordes, A., y Bengio, Y. (2011). Deep sparse rectifier neural networks. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 15:9.

Goldberg, Y. y Levy, O. (2014). word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv:1402.3722 [cs, stat]*.

Goodfellow, I., Bengio, Y., y Courville, A. (2016). *Deep Learning*. The MIT Press.

Goose, A. (1975). *La néologie française aujourd'hui*. Lngue et Langage. Conseil International de la Langue Française.

Gross, G. (1994). Classes d'objets et description des verbes. *Langages*, 28(115):15–30. http://dx.doi.org/10.3406/lgge.1994.1684.

Guerrero Ramos, G. (2015). Uso de neologismos recogidos y propagados por la prensa. *Neologica*, 1(9):223–249.

Guerrero-Ramos, G. y Pérez-Lagos, M. F. (2012). ¿Es la composición culta, en la actualidad, el procedimiento más productivo para la creación de neologismos? *Terminàlia*, pp. 26–36–36.

Guespin, L. (1991). La circulation terminologique et les rapports entre science, technique et production. *Cahiers de Linguistique Sociale*, 1(18):59–78.

Guilbert, L. (1971). La Néologie Scientifique et Technique. *La Banque des Mots*, 1(1):45–54.

Guilbert, L. (1973). Théorie du néologisme. *Cahiers de l'Association internationale des études francaises*, 25(1):9–29. http://dx.doi.org/10.3406/caief.1973.1020.

Guilbert, L. (1975). *La créativité lexicale*. Langue et Langage. Larousse Université.

Gulordava, K. y Baroni, M. (2011). A distributional similarity approach to the detection of semantic change in the google books ngram corpus. En *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, GEMS '11, pp. 67–71, Stroudsburg, PA, USA. Association for Computational Linguistics.

Gurney, K. (1997). *An Introduction to Neural Networks*. Taylor & Francis, Inc., Bristol, PA, USA.

Hadamard, J. (1908). *Mémoire sur le problème d'analyse relatif à l'équilibre des plaques élastiques encastrées, par M. Jacques Hadamard.* Impr. Nationale, París.

Hanks, P. (2004). The syntagmatics of metaphor and idiom. *International Journal of Lexicography*, 17(3):245–274.

Hassoun, M. H. (1995). *Fundamentals of Artificial Neural Networks*. MIT Press.

Hastie, T., Tibshirani, R., y Friedman, J. (2017). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Series in Statistics. Springer, second edition edición.

Haykin, S. S. (2009). *Neural networks and learning machines*. Prentice Hall, New York, 3rd ed edición.

Hinton, G. E., Osindero, S., y Teh, Y.-W. (2006). A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.*, 18(7):1527–1554. http://dx.doi.org/10.1162/neco.2006.18.7.1527.

Hinton, G. E. y Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507. http://dx.doi.org/10.1126/science.1127647.

Ho, T. K. (1995). Random decision forests. En *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volumen 1, pp. 278–282 vol.1. http://dx.doi.org/10.1109/ICDAR.1995.598994.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844. http://dx.doi.org/10.1109/34.709601.

Holz, F. y Teresniak, S. (2010). Towards automatic detection and tracking of topic change. En Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing*, pp. 327–339, Berlín, Heidelberg. Springer Berlin Heidelberg.

Hu, W., Zhang, J., y Zheng, N. (2016). Different Contexts Lead to Different Word Embeddings. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 762–771.

Huang, E. H., Socher, R., Manning, C. D., y Ng, A. Y. (2012). Improving Word Representations via Global Context and Multiple Word Prototypes. En *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pp. 873–882, Stroudsburg, PA, USA. Association for Computational Linguistics.

Huffman, D. A. (1952). A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101. http://dx.doi.org/10.1109/JRPROC.1952.273898.

Janssen, M. (2005a). NeoTrack: semi-automatic neologism detection. En *APL XXI*, Porto, Portugal.

Janssen, M. (2005b). Open source lexical information network. En *Third International Workshop on Generative Approaches to the Lexicon*, Geneva, Suiza.

Janssen, M. (2009). Detección de Neologismos: una perspectiva computacional. *Debate Terminológico*, 5(05):68–75.

Janssen, M. (2012). NeoTag: a POS Tagger for Grammatical Neologism Detection. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pp. 2118–2124.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., y Mikolov, T. (2016a). FastText.zip: Compressing text classification models. *arXiv:1612.03651 [cs]*.

Joulin, A., Grave, E., Bojanowski, P., y Mikolov, T. (2016b). Bag of Tricks for Efficient Text Classification. *arXiv:1607.01759 [cs]*.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., y Wu, Y. (2016). Exploring the Limits of Language Modeling. *arXiv:1602.02410 [cs]*.

Jurafsky, D. y Martin, J. H. (2009). *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*. Pearson Prentice Hall, Upper Saddle River, N.J.

Kantorovich, L. V. y Akilov, G. P. (1982). *Functional analysis*. Pergamon Press, Oxford; New York, 2d ed edición.

Kekeç, T., van der Maaten, L., y Tax, D. M. J. (2018). PAWE: Polysemy Aware Word Embeddings. En *Proceedings of the 2nd International Conference on Information System and Data Mining - ICISDM '18*, pp. 7–13, Lakeland, FL, USA. ACM Press. http://dx.doi.org/10.1145/3206098.3206101.

Kiefer, J. y Wolfowitz, J. (1952). Stochastic Estimation of the Maximum of a Regression Function. *The Annals of Mathematical Statistics*, 23(3):462–466. http://dx.doi.org/10.1214/aoms/1177729392.

Kingma, D. P. y Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*.

Kleinberg, E. M. (1990). Stochastic discrimination. *Annals of Mathematics and Artificial Intelligence*, 1(1):207–239. http://dx.doi.org/10.1007/BF01531079.

Kleinberg, E. M. (1996). An overtraining-resistant stochastic modeling method for pattern recognition. *The Annals of Statistics*, 24(6):2319–2349. http://dx.doi.org/10.1214/aos/1032181157.

Kobayashi, M. y Takeda, K. (2000). Information retrieval on the web. *ACM Computing Surveys*, 32(2):144–173. http://dx.doi.org/10.1145/358923.358934.

Lau, J. H., Baldwin, T., y Newman, D. (2013). On collocations and topic models. *ACM Trans. Speech Lang. Process.*, 10(3):10:1–10:14. http://dx.doi.org/10.1145/2483969.2483972.

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

Lebret, R. P. (2016). *Word Embeddings for Natural Language Processing*. Tesis doctoral, École Polytechnique Fédérale de Lausanne.

LeCun, Y., Bengio, Y., y Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444. http://dx.doi.org/10.1038/nature14539.

Leduc-Adine, J.-P. (1980). De la terminologie grammaticale: quelques problèmes théoriques et pratiques. *Langue française*, 47(1):6–24. http://dx.doi.org/10.3406/lfr.1980.5058.

Legrand, S., Tyrväinen, P., y Saarikoski, H. (2003). Bridging the word disambiguation gap with the help of owl and semantic web ontologies. En *Workshop on Ontologies and Information Extraction, Eurolan*.

Li, G. y Wang, H. (2014). Improved Automatic Keyword Extraction Based on TextRank Using Domain Knowledge. En *Natural Language Processing and Chinese Computing*, pp. 403–413. Springer. http://dx.doi.org/10.1007/978-3-662-45924-9_36.

Liu, D. C. y Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528. http://dx.doi.org/10.1007/BF01589116.

Liu, P., Qiu, X., y Huang, X. (2015). Learning Context-Sensitive Word Embeddings with Neural Tensor Skip-Gram Model. *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pp. 1284–1290.

Louviere, J. J., Flynn, T. N., y Marley, A. A. J. (2015). *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge University Press. http://dx.doi.org/10.1017/CBO9781107337855.

Luhn, H. P. (1957). A Statistical Approach to Mechanized Encoding and Searching of Literary Information. *IBM Journal of Research and Development*, 1(4):309–317. http://dx.doi.org/10.1147/rd.14.0309.

Mancini, M., Camacho-Collados, J., Iacobacci, I., y Navigli, R. (2017). Embedding Words and Senses Together via Joint Knowledge-Enhanced Training. En *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pp. 100–111, Vancouver, Canadá. Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/K17-1012.

Manning, C. D., Raghavan, P., y Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.

Manning, C. D., Raghavan, P., y Schütze, H. (2009). Web crawling and indexes. *Introduction to Information Retrieval*, pp. 443–459.

Manwar, A. B., Mahalle, H. S., y Chinchkhede, K. D. (2012). A vector space model for information retrieval: a matlab approach. *Indian Journal of Computer Science and Engineering*, 3(2):222–229.

Marley, A. y Louviere, J. (2005). Some probabilistic models of best, worst, and best–worst choices. *Journal of Mathematical Psychology*, 49(6):464 – 480. https://doi.org/10.1016/j.jmp.2005.05.003.

McCulloch, W. S. y Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Mathematical Biophysics*, 5:115–133.

Mejri, S. (2006). La reconnaissance automatique des néologismes de sens. En Daniel Blampain, Philippe Thoiron, M. V. C., editor, *Septièmes Journées scientifiques du réseau LTT, 8-10 septembre 2005*, pp. 545–557, Bruxelles, Bélgica. AUF.

Mejri, S. (2010). Néologie et traitement automatique. En Cabré, M. T., editor, *I Congrés International de Neologia de les Llengües Romàniques*, pp. 99–110, Barcelona. Institut Universitari de Lingüística Aplicada.

Michie, D., Spiegelhalter, D. J., y Taylor, C. C. (1994). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, Upper Saddle River, NJ, USA.

Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. En *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. http://dx.doi.org/10.3115/1219044.1219064.

Mihalcea, R. (2005). Language Independent Extractive Summarization. En *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*.

Mihalcea, R. y Tarau, P. (2004). TextRank: Bringing Order into Text. En *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

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

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

Mikolov, T., Yih, W.-t., y Zweig, G. (2013c). Linguistic regularities in continuous space word representations. En *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 746–751. Association for Computational Linguistics.

Minsky, M. L. y Papert, S. A. (1988). *Perceptrons: Expanded Edition*. MIT Press, Cambridge, MA, USA.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., y Hassabis, D. (2015). Human-level Control Through Deep Reinforcement Learning. *Nature*, 518(7540):529–533. `http://dx.doi.org/10.1038/nature14236`.

Moeschler, J. (1974). Aspects de la néologie sémantique. *Langages*, 8(36):6–19. `http://dx.doi.org/10.3406/lgge.1974.2270`.

Molina, A., Sierra, G., y Torres-Moreno, J.-M. (2010). La energía textual como medida de distancia en agrupamiento de definiciones. En *Proceedings of the 10th Journées Internationales d'Analyse statistique des Données Textuelles*, volumen 3, pp. 215–226.

Morel Santasusagna, J., Torner, S., Vivaldi, J., Cabré, M. T., y Yzaguirre, L. d. (1998). *El Corpus de l'IULA: etiquetaris*. Universitat Pompeu Fabra, Barcelona.

Müller, A. C. y Guido, S. (2017). *Introduction to machine learning with Python a guide for data scientists*. O'Reilly.

Nair, V. y Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. *ICML'10 Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 807–814.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2):10:1–10:69. `http://dx.doi.org/10.1145/1459352.1459355`.

Nazar, R. (2010). *A Quantitative Approach to Concept Analysis*. Tesis doctoral, Universitat Pompeu Fabra. Institut Universitari de Lingüística Aplicada.

Nazar, R. (2011). Neología semántica: un enfoque desde la lingüística cuantitativa.

Nazar, R. (2013). Word sense discrimination using statistic analysis of texts. *Barcelona Investigación Arte Creación*, 1(1):5–26.

Nazar, R. y Cabré, M. T. (2012). Supervised learning algorithms applied to terminology extraction. En *Proceedings of TKE 2012 (Terminology and Knowledge Engineering) Conference*, Madrid.

Nazar, R., Vivaldi, J., y Wanner, L. (2007). Towards quantitative concept analysis. *Procesamiento del Lenguaje Natural*, 1(39).

Nilsson, J., N. (1996). *Introduction to machine learning: An early draft of a proposed textbook*. Stanford University, USA.

Opitz, D. y Maclin, R. (1999). Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, 11:169–198. http://dx.doi.org/10.1613/jair.614.

Page, L., Brin, S., Motwani, R., y Winograd, T. (1999). The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab.

Patwardhan, S., Banerjee, S., y Pedersen, T. (2003). Using Measures of Semantic Relatedness for Word Sense Disambiguation. *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, 4:241–257. http://dx.doi.org/10.1007/3-540-36456-0{_}24.

Pay, T. (2016). Totally automated keyword extraction. En *2016 IEEE International Conference on Big Data (Big Data)*, pp. 3859–3863. IEEE. http://dx.doi.org/10.1109/BigData.2016.7841059.

Pay, T. y Lucci, S. (2017). Automatic keyword extraction: An ensemble method. *Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017*, 2017-Janua(December 2017):4816–4818. http://dx.doi.org/10.1109/BigData.2017.8258552.

Pay, T., Lucci, S., y Cox, J. (2018). An Ensemble of Automatic Keyphrase Extractors: TextRank , RAKE and TAKE. http://dx.doi.org/10.13140/RG.2.2.13961.70243/1.

Pearson, K. (1895). VII. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58(347-352):240–242. http://dx.doi.org/10.1098/rspl.1895.0041.

Pecina, P. (2009). *Lexical Association Measures. Collocation Extraction*. Ústav Formální A Aplikované Lingvistiky.

Pennington, J., Socher, R., y Manning, C. D. (2014). Glove: Global vectors for word representation. En *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543.

Picoche, J. (1977). *Précis de lexicologie française*. Nathan, París.

Pilehvar, M. T., Camacho-Collados, J., Navigli, R., y Collier, N. (2017). Towards a Seamless Integration of Word Senses into Downstream NLP Applications. En *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1857–1869, Vancouver, Canadá. Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/P17-1170.

Pilehvar, M. T. y Collier, N. (2016). De-Conflated Semantic Representations. En *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1680–1690, Austin, Texas. Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/D16-1174.

Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45. http://dx.doi.org/10.1109/MCAS.2006.1688199.

Renau, C. y Nazar, R. (2011). Análisis cuantitativo del uso real de los verbos pronominales estrictos del castellano utilizando un corpus diacrónico (google books). En *Actas del III Congreso Internacional de Lingüística de Corpus, AELINCO*, pp. 287–298, Valencia.

Renouf, A. (1998). Aviating among the hapax legomena: morphological grammaticalisation in current british newspaper english. En Renouf, A. y Baayen, R. H., editores, *Explorations in Corpus Linguistics*. Rodopi.

Renouf, A. (2010). Identification automatique de la néologie lexicologique et sémantique: questions soulevées par notre méthode. En Cabré, M. T., Domènech, O., Estopà, R., Freixa, J., y Lorente, M., editores, *Actes del Congrés Internacional de Neologia de les Llengües Romàniques*, pp. 129–141, Barcelona. Intitut Universitari de Lingüística Aplicada.

Renouf, A. (2012). A finer definition of neology in English: the life-cycle of a word. *Corpus perspectives on patterns of lexis*, pp. 177–208.

Reutenauer, C., Jacquey, E., y Ollinger, S. (2011). Neologismes de sens: contribution à leur caracterisation dans un corpus autour du th àme de la crise financière. En *II Congrès International de Néologie des Langues Romanes (CINEO2011)*, Sao Paulo, Brasil.

Rey, A. (1976). Néologisme, un pseudo-concep? *Cahiers de lexicologie*, 28(1):3–17.

Rey, A. (1995). *Essays on Terminology*. John Benjamins Publishing. http://dx.doi.org/10.1075/btl.9.

Rey, A. (2005). The concept of neologism and the evolution of terminologies in individual languages. *Terminology*, 11(2):311–331.

Richardson, M. y Domingos, P. (2002). The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank. *Proceedings of Advances in Neural Information Processing Systems*, 14:8.

Robbins, H. y Monro, S. (1951). A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400–407. http://dx.doi.org/10.1214/aoms/1177729586.

Rohrdantz, C., Hautli, A., Mayer, T., Butt, M., Keim, D. A., y Plank, F. (2011). Towards tracking semantic change by visual analytics. En *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pp. 305–310, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rondeau, G. (1984). *Introduction à la terminologie*. Gaëtan Morin, Québec, Canadá.

Rose, S., Engel, D., Cramer, N., y Cowley, W. (2010). Automatic Keyword Extraction from Individual Documents. En *Text Mining: Applications and Theory*, pp. 1–20. John Wiley & Sons, Ltd, Chichester, UK. http://dx.doi.org/10.1002/9780470689646.ch1.

Rosen, Z. (2018). Computationally Constructed Concepts: A Machine Learning Approach to Metaphor Interpretation Using Usage-Based Construction Grammatical Cues. En *Proceedings of the Workshop on Figurative Language Processing*, pp. 102–109, New Orleans, Louisiana. Association for Computational Linguistics. http://dx.doi.org/10.18653/v1/W18-0912.

Rosenblatt, F. (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain. *Psychological Review*, pp. 65–386.

Rosenblatt, F. (1962). *Principles of neurodynamics: perceptrons and the theory of brain mechanisms.* Spartan Books, Washington.

Rumelhart, D. E., Hinton, G. E., y Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536. http://dx.doi.org/10.1038/323533a0.

Sablayrolles, J.-F. (1996). Néologismes: pour une typologie des typologies. *Cahiers du CIEL (1996–1997)*, 2(69):11–48.

Sablayrolles, J.-F. (2000). *La Néologie en français contemporain: examen du concept et analyse de productions néologiques récentes*. Honoré Champion, París.

Sablayrolles, J.-F. (2006). La néologie aujourd'hui. En Gruaz, C., editor, *A la recherche du mot: De la langue au discours*, pp. 141–157. Lambert-Lucas.

Sagi, E., Kaufmann, S., y Clark, B. (2009). Semantic density analysis: Comparing word meaning across time and phonetic space. En *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, GEMS '09, pp. 104–111, Stroudsburg, PA, USA. Association for Computational Linguistics.

Salakhutdinov, R. y Hinton, G. (2009). Deep Boltzmann Machines. En van Dyk, D. y Welling, M., editores, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volumen 5 de *Proceedings of Machine Learning Research*, pp. 448–455, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA. PMLR.

Salton, G. y Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523. http://dx.doi.org/10.1016/0306-4573(88)90021-0.

Salton, G., Fox, E. A., y Wu, H. (1983). Extended Boolean Information Retrieval. *Commun. ACM*, 26(11):1022–1036. http://dx.doi.org/10.1145/182.358466.

Salton, G. y McGill, M. (1984). *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company.

Salton, G., Wong, A., y Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620. http://dx.doi.org/10.1145/361219.361220.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117. http://dx.doi.org/10.1016/j.neunet.2014.09.003.

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

Shuyo, N. (2010). Language detection library for java.

Singhal, A. (2001). Modern Information Retrieval: A Brief Overview. *IEEE Data Engineering Bulletin*, 24.

Sparck-Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*.

Strang, G. (2006). *Linear algebra and its applications*. Thomson, Brooks/Cole, Belmont, CA.

Sun, Y., Rao, N., y Ding, W. (2017). A Simple Approach to Learn Polysemous Word Embeddings. *arXiv:1707.01793 [cs]*.

Tan, P.-N., Steinbach, M., y Kumar, V. (2005). *Introduction to Data Mining, (first Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Tanasescu, C., Kesarwani, V., e Inkpen, D. (2018). Metaphor Detection by Deep Learning and the Place of Poetic Metaphor in Digital Humanities. *The Thirty-First International Florida Artificial Intelligence Research Society Conference (FLAIRS-31)*, pp. 122–127.

Taule, M., Martı, M. A., y Recasens, M. (2004). AnCora: Multilevel Annotated Corpora for Catalan and Spanish. *Proceedings of 6th International Conference on language Resources and Evaluation*.

Tebé, C. (2002). Bases pour une sélection de neologismes. En M. Teresa Cabré, Judit Freixa, E. S., editor, *Lèxic i neologia*, pp. 43–50. Observatori de Neologia and Universitat Pompeu Fabra, Barcelona.

Temmerman, R. (1997). Questioning the univocity ideal. The difference between socio-cognitive Terminology and traditional Terminology. *HERMES - Journal of Language and Communication in Business*, 10(18):51. http://dx.doi.org/10.7146/hjlcb.v10i18.25412.

Temmerman, R. (2000). *Towards new ways of terminology description: the sociocognitive-approach*. Número 3 en Terminology and lexicography research and practice. Benjamins, Amsterdam.

Temmerman, R. (2001). Sociocognitive terminology theory. En Cabré, M. T. y Feliu, J., editores, *Terminología y Cognición*, pp. 75–92. Universitat Pompeu Fabra. Institut Universitari de Lingüística Aplicada, Barcelona.

Tieleman, T. y Hinton, G. (2012). Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning.

Torres, A. (2015). Herramientas de detección y extracción de neología: estado de la cuestión. Tesis de máster, Universitat Pompeu Fabra.

Torres-Moreno, J.-M. (1997). *Machine learning and generalization by Neural Networks: new constructive algorithms*. Theses, Institut National Polytechnique de Grenoble - INPG.

Torres-Moreno, J. M. (2011). *Résumé automqtique de documents - une approche statistique*. Hermès-Lavoisier.

Torres-Moreno, J. M. (2014). *Automatic Text Summarization*. Wiley & Sons.

Torres-Moreno, J. M., Velázquez-Morales, P., y Meunier, J.-G. (2001). Cortex: un algorithme pour la condensation automatique de textes. En *ARCo'2001*.

Torres-Moreno, J. M., Velázquez-Morales, P., y Meunier, J.-G. (2002). Condensés de textes par des méthodes numériques. En *JADT 2002: 6es Journées Internationales d'Analyse Statistique Des Données Textuelles Combine*.

Tournier, J. (1985). *Introduction descriptive à la lecigénétique de l'anglais contemporain*. Champion-Slatkine, París; Geneva.

Tournier, J. (1991). *Précis de lexicologie anglaise*. Nathan.

Trask, A., Michalak, P., y Liu, J. (2015). sense2vec - a fast and accurate method for word sense disambiguation in neural word embeddings. *CoRR*, abs/1511.06388.

Tsvetkov, Y., Boytsov, L., Gershman, A., Nyberg, E., y Dyer, C. (2014). Metaphor Detection with Cross-Lingual Model Transfer. En *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 248–258, Baltimore, Maryland. Association for Computational Linguistics. `http://dx.doi.org/10.3115/v1/P14-1024`.

Van Der Maaten, L. (2009). Learning a Parametric Embedding by Preserving Local Structure. En *Artificial Intelligence and Statistics*, pp. 384–391.

Van Der Maaten, L. (2014). Accelerating t-SNE Using Tree-based Algorithms. *Journal of Machine Learning Research*, 15(1):3221–3245.

Van Der Maaten, L. y Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Van Der Maaten, L. y Hinton, G. (2012). Visualizing non-metric similarities in multiple maps. *Machine Learning*, 87(1):33–55. http://dx.doi.org/10.1007/s10994-011-5273-4.

Vapnik, V. (1999). An overview of statistical learning theory. *IEEE Trans. Neural Networks*, 10(5):988–999. http://dx.doi.org/10.1109/72.788640.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer New York, New York, NY.

Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., y Trón, V. (2007). Parallel corpora for medium density languages. En Nicolov, N., Bontcheva, K., Angelova, G., y Mitkov, R., editores, *Current Issues in Linguistic Theory*, volumen 292, pp. 247–258. John Benjamins Publishing Company, Amsterdam. http://dx.doi.org/10.1075/cilt.292.32var.

Veale, T., Shutova, E., y Klebanov, B. B. (2016). Metaphor: A Computational Perspective. *Synthesis Lectures on Human Language Technologies*, 9(1):1–160. http://dx.doi.org/10.2200/S00694ED1V01Y201601HLT031.

Verhulst, P.-F. (1845). Recherches Mathématiques sur la Loi d'Accroissement de la Population. En *Noveaux Mémoires de l'Académie Royale des Sciences et Belles-Lettres de Bruxelles*, número XVIII en Mémoires de l'Académie. L'Académie Royale de Bruxelles et de l'Université Louvain.

Vivaldi, J. (2003). *Sistema de extracción de candidatos a término. YATE. Manual de utilización*. Institut Universitari de Lingüística Aplicada, Universitat Pompeu Fabra, Barcelona.

Vivaldi, J. (2009). Corpus And Exploitation Tool: IULACT and Bwananet. En *A survey on corpus-based research = Panorama de investigaciones basadas en corpus*, pp. 224–239, Murcia. Asociación Española de Lingüística del Corpus.

Vonk, E., Jain, L., y Veelenturf, L. (1995). Neural network applications. En *Proceedings of the International Conference and Workshops ETD2000, Electreonic Technology Directions to the year 2000*, United States. IEEE. http://dx.doi.org/10.1109/ETD.1995.403490.

Walker, S. H. y Duncan, D. B. (1967). Estimation of the Probability of an Event as a Function of Several Independent Variables. *Biometrika*, 54(1/2):167–179. http://dx.doi.org/10.2307/2333860.

Walter, H. (1984). L'innovation lexicale chez les jeunes Parisiens. *La Linguistique*, 20:69–84.

Widrow, B. y Lehr, M. A. (1993). Artificial neural networks of the perceptron, madaline and backpropagation family. En *Neurobionics*, pp. 133–205. Elsevier. http://dx.doi.org/10.1016/B978-0-444-89958-3.50013-9.

Wu, H. C., Luk, R. W. P., Wong, K. F., y Kwok, K. L. (2008). Interpreting TF-IDF term weights as making relevance decisions. *ACM Transactions on Information Systems*, 26(3):1–37. http://dx.doi.org/10.1145/1361684.1361686.

Wüster, E. (1973). Was ist angewandte Sprach- wissenschaft? Ein Wegweiser durch das Dickicht der Terminologien wird immer notwendiger. *Wiener Zeitung*, 148.

Wüster, E. (1979). *Einführung in die allgemeine Terminologielehre und terminologische Lexikographie*. Springer, Viena.

Wüster, E. (1998). *Introducción a la teoría general de la terminología y a la lexicografía terminológica*. Institut Universitari de Lingüística Aplicada, Barcelona.

Xu, L., Sun, S., y Wang, Q. (2016). Text similarity algorithm based on semantic vector space model. En *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pp. 1–4. IEEE. http://dx.doi.org/10.1109/ICIS.2016.7550928.

Yin, Z. y Shen, Y. (2018). On the dimensionality of word embedding. En Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., y Garnett, R., editores, *Advances in Neural Information Processing Systems 31*, pp. 887–898. Curran Associates, Inc.

Zaki, M. J. y Meira, Jr, W. (2014). *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press, 1 edición. http://dx.doi.org/10.1017/CBO9780511810114.