Desarrollo de un sistema de aprendizaje automático supervisado para la desambiguación léxica automática utilizando DAMIEN (Data Mining Encountered)

Development of a Supervised Machine Learning System for Automatic Word Sense Disambiguation using DAMIEN (Data Mining Encountered)

FREDY NUÑEZ TORRES
UNIVERSIDAD CATÓLICA DE CHILE
MARÍA BEATRIZ PÉREZ CABELLO DE ALBA
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

Uno de los mayores desafíos que se nos presentan a la hora de acometer tareas relacionadas con el procesamiento del lenguaje natural y, en particular, con el tratamiento de recursos lingüísticos informatizados, es la ambigüedad léxica. En este trabajo abordamos el tratamiento de la desambiguación léxica dentro del entorno informático DAMIEN (*Data Mining ENcountered*), una herramienta que integra técnicas de múltiples disciplinas dentro de análisis de texto (i.e. lingüística de corpus, estadística y minería textual) para ayudar en tareas de investigación lingüística (i.e. recolección de datos, extracción de información, clasificación de textos, entre otras). A modo de experimento ilustrativo, llevamos a cabo un estudio de las unidades léxicas polisémicas "cabeza", "cara" y "carta", y presentamos los resultados del sistema de desambiguación automática desarrollado con la herramienta DAMIEN. Dentro de los modelos que ofrece el entorno, hemos elegido el método de aprendizaje automático supervisado mediante algoritmo bayesiano ingenuo por tratarse del método que mejores resultados ha dado para la desambiguación léxica automática. Se trata de un modelo matemático que consiste en extraer información de un corpus a partir de conjuntos de datos previamente etiquetados (corpus de entrenamiento) para que la máquina pueda clasificar automáticamente conjuntos de datos nuevos (corpus de prueba). Es importante resaltar la flexibilidad y riqueza del entorno DAMIEN tanto para el tratamiento de recursos lingüísticos informatizados como para el montaje de experimentos del procesamiento del lenguaje natural.

Palabras clave: *lingüística computacional; procesamiento del lenguaje natural; lingüística de corpus; ambigüedad léxica; aprendizaje automático*

Word sense ambiguity is one of the major challenges we face when we carry out tasks related to Natural Language Processing, in particular those related to the processing of electronic language resources. In this study we address word sense disambiguation within the computing environment DAMIEN (*Data Mining ENcountered*). DAMIEN is an online workbench that embeds several techniques from different fields (corpus linguistics, statistics and text mining) in order to deal with text analysis to help in linguistic research tasks such as data collection, information retrieval and text classification, among others. By way of experiment, we carry out the analysis of the polysemic lexical units "cabeza", "cara" and "carta" in Spanish and present the results of the automatic disambiguation system developed with DAMIEN. Among the models that the environment offers we have deployed the supervised machine learning method with ingenious bayes algorithm because it has traditionally given the best results for automatic word sense disambiguation. It is a mathematical model that consists in extracting information from a corpus, setting from previously tagged datasets (training corpus), so that new datasets can be automatically classified by the system (trained corpus). It is important to highlight the flexibility and potentialities of DAMIEN for both the processing of electronic linguistic resources and the design of experiments in the field of natural language processing.

Keywords: *computational linguistics; natural language processing; corpus linguistics; lexical ambiguity; machine learning*

1. INTRODUCCIÓN

1.1 *Definición computacional de la ambigüedad léxica*

El fenómeno de la ambigüedad léxica se ha abordado a partir de una aproximación simplificada si se la compara con la descripción lingüística tradicional (Núñez-Torres, 2013). En el ámbito del procesamiento del lenguaje natural (en adelante PLN), los trabajos de Jurafsky y Martin (1998) han sistematizado las convenciones conceptuales con las que se hace referencia a ciertos rasgos de las unidades léxicas. En primer lugar, se define la *forma de palabra*, o morfo (*wordform*), que corresponde a la forma flexionada de una unidad léxica, tal y como aparece en el cotexto, o contexto oracional. En segundo lugar, se ha restringido el uso del concepto *palabra* a la noción de palabra ortográfica (también llamada *palabra gráfica*), por lo que un *n-grama* corresponde a una secuencia de *n* palabras. Así, utilizaremos el concepto de *unidad léxica* para hacer referencia a una unidad funcional de significado que se realiza lingüísticamente mediante una o más palabras.

Así, el criterio que hemos establecido para reconciliar las definiciones de los conceptos de *palabra* y *unidad léxica* implica que todas las unidades léxicas están compuestas por n-gramas, pero no todos los n-gramas pueden considerarse unidades léxicas; es decir, los n-gramas, en este caso sintácticos, se pueden definir como una secuencia de unidades léxicas de extensión *n*. Según lo anterior, una secuencia como "Miguel", donde *n=1* corresponde a un unigrama; una secuencia como "El Quijote", donde *n=2*, corresponde a un bigrama; y una secuencia como "Miguel de Cervantes", donde *n=3*, corresponde a un trigrama.

En tercer lugar, se establece como lema la forma citada que presenta la misma raíz de una forma de palabra. Por ejemplo, la unidad léxica "bancos" corresponde a un morfo, mientras que "banco" es su lema.[1]

Por otra parte, el sentido corresponde a una representación discreta del significado de una unidad léxica. Finalmente, la ambigüedad léxica se puede manifestar en dos casos, la homonimia y la polisemia, que definimos y ejemplificamos en (1) y (2) respectivamente.

(1) Homonimia: coincidencia de dos lemas en su escritura o pronunciación, aun cuando difieren en sus sentidos. Presenta, a su vez, dos tipos:

   a. Homógrafo: coincidencia gráfica en la escritura
      p. ej. *lengua* (idioma-órgano de la boca)

   b. Homófono: coincidencia en la pronunciación, diferencia en la escritura
      p. ej. *rallar* (desmenuzar) vs. *rayar* (hacer líneas)

(2) Polisemia: coincidencia de dos lemas en su escritura y pronunciación, cuyos sentidos se encuentran relacionados con el mismo significado:

[1] El tratamiento de estos conceptos, desde el ámbito de la lexicografía, establece que una unidad léxica es seleccionada como lema, y se posiciona por tanto como el criterio desde el que se organizan las definiciones en los diccionarios semasiológicos.

a. p. ej. *cubo* (recipiente) vs. *cubo* (figura geométrica)

1.2 *El aporte de las técnicas de PLN en las ciencias del lenguaje*

Desde los inicios de los estudios en inteligencia artificial ha existido una tensión en cuanto a la integración de las disciplinas de la informática y la lingüística. La informática ha propiciado esencialmente el desarrollo de modelos estocásticos (o probabilísticos), que se caracterizan por la aplicación de técnicas matemáticas sobre un gran volumen de datos textuales con el objetivo de inferir conocimiento lingüístico (Espunya i Prat, 1994; Allen, 1995; Cantos-Gómez, 1996). Estas implementaciones no almacenan conocimiento lingüístico (o conceptual), sino que aplican determinadas técnicas matemáticas sobre corpus textuales con el fin de extraer conocimiento. Así, estos sistemas estocásticos son capaces de inferir conocimiento lingüístico a través de la utilización de algoritmos. En definitiva, se trata de una construcción automatizada del conocimiento, basada en una aproximación computacional al análisis de textos como volúmenes de información computable.

Este tipo de modelos consideran las lenguas naturales como un conjunto de sucesos que presentan una determinada frecuencia. Esto quiere decir que cada unidad lingüística—sea un morfema, palabra, sintagma o cualquier tipo de categoría, ya sea morfológica o sintáctica—presenta una probabilidad específica de manifestarse o aparecer en un contexto oracional acotado. Así, dado que esta perspectiva depende de la calidad y cantidad de la información almacenada en un corpus lingüístico, mientras mayor sea el número de datos utilizados, mejor se comportará el modelo, cualquiera sea este. Un ejemplo representativo y bastante actual de la aplicación de los métodos estadísticos son los modelos basados en el aprendizaje automático (*machine learning*). Se trata de un ámbito de estudio de la inteligencia artificial cuyo objetivo es el desarrollo de algoritmos que sean capaces de representar eficientemente determinados conjuntos de datos (Choi, Coyner, Kalpathy-Cramer, Chiang y Campbell, 2020). Específicamente, se espera poder determinar automáticamente la probabilidad de que a una unidad lingüística se le asigne un valor predefinido como correcto. Los métodos de aprendizaje automático que se utilizan con mayor frecuencia son: a) supervisado; b) no supervisado; y c) profundo.

El *aprendizaje supervisado* determina patrones en un corpus de entrenamiento, con el objetivo de mapear atributos que servirán como conjunto de datos a partir de los cuales realizar predicciones en un nuevo corpus. Entonces, se le denomina supervisado porque el modelo es capaz de inferir información a partir de un algoritmo y un conjunto de datos previamente etiquetado, y transferir sus características a una predicción (Moor, 2006).

En el caso del *aprendizaje no supervisado*, la detección de patrones en un conjunto de datos se realiza a través de un algoritmo sin la necesidad de haber incorporado información previa. Esta técnica se emplea predominantemente para las tareas de agrupación, asociación y detección de anomalías (Hastie, Tibshirani y Friedman, 2009; James, Witten, Hastie y Tibshirani, 2013).

El *aprendizaje profundo* es un conjunto de técnicas basadas en la representación matemática de redes neuronales. Un ejemplo actualizado de este método aplicado al PLN es el de la incrustación de palabras (*word embeddings*), que consiste en determinar la probabilidad condicional de que una secuencia de palabras determinadas aparezca en un texto, mediante la asignación de un vector numérico a cada palabra. Así, un valor de probabilidad alto o mayor indicará que la secuencia en análisis se utiliza de manera más frecuente (Li y Yang, 2017).

En síntesis, para el PLN, el problema de la desambiguación léxica automática se ha estado abordando, y con relativo éxito, desde la década de 1980. Sin embargo, aunque se

trate de un problema antiguo, aún existe un campo de desarrollo relevante en el que se proponen distintas maneras de mejorar el proceso de automatización. Por otra parte, para la lingüística, es el problema mismo de la desambiguación léxica automática lo que aporta una perspectiva novedosa desde este ámbito aplicado e interdisciplinar, puesto que los modelos teóricos que son capaces de describir y explicar el fenómeno ya son bastante satisfactorios, aunque inaplicables por sí solos computacionalmente.

Este ámbito interdisciplinar supone una perspectiva altamente valiosa, puesto que, si bien los métodos estocásticos son muy eficientes para tareas de PLN, no necesariamente estarían abordando la reproducción de los patrones con los que funciona la mente humana. Por el contrario, los modelos probabilísticos no representan la manifestación de una capacidad cognitiva humana como el lenguaje, sino que en realidad corresponden a un conjunto de métodos que facultan a las máquinas para examinar las lenguas naturales, con el objetivo de imitar, y no reproducir, la capacidad humana de comprender el lenguaje. Esta paradoja, en definitiva, es una motivación para el trabajo colaborativo tanto de lingüistas como de ingenieros informáticos, cuyos objetivos de investigación comunes nos permitirán comprender de manera más exhaustiva la mente humana, en el esfuerzo de reproducir artificialmente sus habilidades cognitivas. Un ejemplo de esto es, sin dudas, el problema de la desambiguación léxica automática.

1.2 *El entorno de trabajo DAMIEN (Data Mining ENcountered) para experimentos de análisis de corpus*

DAMIEN (*DAta MIning ENcountered*)[2] es un entorno informático que puede integrar técnicas de múltiples disciplinas dentro de análisis de texto (i.e. lingüística de corpus, estadística y minería textual) para apoyar la investigación lingüística de manera más efectiva. A continuación, se realiza una descripción de los componentes y características más relevantes de DAMIEN, según Periñán-Pascual (2017), que serán relevantes para la implementación de los experimentos propuestos más adelante. DAMIEN contiene cuatro interfaces especializadas en tareas específicas:

1) *Corpus*: tareas relacionadas con la exploración, preprocesamiento y procesamiento de un corpus o colección de textos.

2) *Statistics*: tareas relacionadas con la descripción e interpretación de datos a partir de la aplicación de parámetros estadísticos.

3) *Mining*: tareas relacionadas con minería de datos y métodos de predicción (como clasificadores o métodos de agrupamiento para el aprendizaje automático).

4) *Evaluation*: tareas relacionadas con la aplicación de medidas de evaluación.

En el ámbito del PLN en general, y el tratamiento de recursos lingüísticos informatizados en particular, DAMIEN propicia que los lingüistas puedan alcanzar sus objetivos de investigación de manera más efectiva mediante la integración de métodos y técnicas provenientes de varios campos dentro del análisis de datos textuales. Actualmente, los programas computacionales disponibles de manera gratuita, como

[2] Disponible, previo registro, en http://www.fungramkb.com/nlp.aspx

TextSTAT[3] y *AntConc*[4] para lingüística de corpus, *R commander*[5] para estadística, *WEKA*[6] para minería de datos o *GATE*[7] para ingeniería lingüística, no son capaces de reunir en una sola interfaz gráfica de usuario todos los requerimientos necesarios para la investigación basada en corpus lingüísticos. Por el contrario, DAMIEN logra integrar en un mismo entorno de trabajo las diferentes herramientas y técnicas que pueden ser aplicadas en el análisis de datos textuales basados en corpus. Estas técnicas provienen principalmente de:

1) Lingüística de corpus (i.e. listas de frecuencia, procesamiento de XML (XSL), administración de bases de datos y consultas SQL, búsqueda por expresiones regulares, etc.).

2) Estadística (i.e. estadística descriptiva e inferencial, representación gráfica de datos).

3) Procesamiento del lenguaje natural (i.e. extracción de n-gramas, derivación, análisis morfológico o sintáctico, etiquetado POS, etc.).

4) Minería de textos (i.e. clasificación y métodos de agrupamiento).

En resumen, DAMIEN permite a los investigadores resolver tres grandes grupos de tareas. Estas tareas, a su vez, representan aquellos pasos necesarios para planificar, montar y evaluar experimentos de análisis de corpus:

1) Administración de colecciones de datos: visualización, edición, aleatorización, búsqueda y extracción de información.

2) Análisis de datos: convergencia de la estadística con la lingüística de corpus a través de la estadística descriptiva (medidas de posición y dispersión) y la estadística inferencial (pruebas estadísticas de distinto tipo, como correlación, regresión, o análisis multivariante).

3) Presentación de datos: transferencia a texto y representaciones gráficas a partir del análisis de datos, para la difusión científica en diferentes formatos.

2. METODOLOGÍA

2.1 *Procedimiento en DAMIEN para experimentos de aprendizaje automático*

A continuación, se presenta una explicación del método de aprendizaje automático supervisado basado en el algoritmo bayesiano ingenuo, junto con el procedimiento específico para la ejecución del experimento en DAMIEN. Esta metodología se divide en cuatro subtipos de procedimientos: a) preprocesamiento, b) procesamiento, c) minería textual y d) evaluación. Posteriormente, para cada uno de estos subtipos de procesamiento

[3] Disponible en https://neon.niederlandistik.fu-berlin.de/en/textstat/
[4] Disponible en https://www.laurenceanthony.net/software/antconc/
[5] Disponible en https://socialsciences.mcmaster.ca/jfox/Misc/Rcmdr/
[6] Disponible en https://www.cs.waikato.ac.nz/ml/weka/
[7] Disponible en https://gate.ac.uk

se estableció un número determinado de tareas. Finalmente, para el desarrollo estandarizado de los experimentos se requirió la ejecución de diez tareas de procesamiento con sus respectivas secuencias de pasos en DAMIEN.

2.2 *Aprendizaje automático supervisado*

En el aprendizaje supervisado, se dispone de un corpus de entrenamiento previamente etiquetado con el sentido correspondiente para cada instancia de una palabra objetivo (Raganato, Camacho-Collados y Navigli, 2017). Por ejemplo, en la oración "en la mesa estaba la carta que el mago adivinó", la palabra objetivo corresponderá a la unidad léxica en análisis que presenta ambigüedad, que en este caso sería "carta", y se tendrían en cuenta los sentidos posibles: 'papel escrito que una persona envía a otra con la intención de comunicarse' o 'cartulinas rectangulares con ilustraciones que se utilizan en los juegos de azar.' Luego, las palabras de contenido adyacentes conformarán el cotexto o conjunto de palabras contextuales {*mesa*, *estaba*, *mago*, *adivinó*}.

Posteriormente, los algoritmos de aprendizaje automático se aplican al corpus de entrenamiento con características contextuales extraídas desde instancias del corpus, considerando los sentidos individuales como clases discretas. La mayoría de los métodos supervisados tradicionales tienen, al menos, cuatro fases de aplicación en común:

1) Selección de un conjunto de datos textuales que muestre las diferentes clasificaciones para cada elemento (valores, atributos, características).

2) Identificación de los patrones asociados con cada elemento.

3) Generalización de patrones.

4) Aplicación de patrones para clasificar nuevos elementos no presentes en el conjunto inicial de datos textuales.

2.2.1 *Algoritmo bayesiano ingenuo (Naïve Bayes)*

En cuanto al algoritmo bayesiano ingenuo, este se define como un clasificador probabilístico. El objetivo de este tipo de modelos matemáticos es extraer información a partir de conjuntos de datos previamente etiquetados o entrenados para que la máquina pueda etiquetar automáticamente conjuntos de datos nuevos, o corpus de prueba. Así, el corpus de entrenamiento representa las etiquetas esperadas, mientras que el corpus de prueba es desde el cual se establecen las etiquetas predichas. Este enfoque se denomina *ingenuo* porque la presencia o ausencia de una característica particular no estará relacionada necesariamente con la presencia o ausencia de cualquier otra, dada la variable original.

La aplicación del algoritmo bayesiano ingenuo en la desambiguación léxica automática, implementado por primera vez por Gale, Church y Yarowsky (1992), está basada en un modelo matemático de dependencias entre los sentidos de palabra y un conjunto de características presentadas en un recurso lingüístico informatizado; es decir, cada una de sus características constituye una probabilidad independiente. La afirmación anterior tiene dos consecuencias relevantes: a) la primera es que se ignora la sintaxis y el carácter lineal de las palabras dentro del cotexto, lo que deriva en el llamado modelo de bolsa de palabras; y, b) la segunda es que la presencia de una palabra en esta bolsa es independiente de la presencia de otra, lo que no es cierto en el caso de las lenguas naturales. Sin embargo, a pesar de estos supuestos simplificadores, y como se señala en

los trabajos de Manning y Schütze (1999), se ha demostrado que este modelo es bastante efectivo desde una perspectiva cognitiva, que es adecuado en el caso de un problema relacionado con el PLN.

En efecto, el enfoque para la desambiguación léxica automática en el que se basa el modelo bayesiano ingenuo representa un enfoque teórico relevante en el ámbito del procesamiento estadístico del lenguaje. La idea del clasificador bayesiano en el contexto de la desambiguación léxica automática es observar palabras objetivo alrededor de una bolsa de palabras contigua en una determinada ventana contextual. Así, cada palabra de contenido dentro de la ventana contextual aportará información relevante acerca del sentido de la palabra ambigua. Este algoritmo se utiliza ampliamente debido a su eficiencia y su capacidad para combinar evidencia de una gran cantidad de características (Escudero, Márquez y Rigau, 2000; Aung, Soe y Thein, 2011; Fulmari y Chaldak, 2014; Gamallo, Sotelo y Pichel, 2014; Gosal, 2015). Es aplicable si el estado de cosas del mundo en el que se basa una clasificación se describe como una serie de características o atributos utilizados para la descripción, y que a su vez son condicionalmente independientes.

Finalmente, el proceso de desambiguación se realiza utilizando la regla de decisión de *Bayes*: se calcula la puntuación de cada sentido de una palabra ambigua y decide el sentido más apropiado para una palabra específica en la oración de prueba de la siguiente manera:

$$P(feature) = \frac{P(sense) \times P(sense)}{P(feature)}$$

Según Fulmari y Chaldak (2014), mediante la aplicación del supuesto de ingenuidad, el algoritmo se reduce a:

$$argmax_{S_1 \in senses(w)}\, P\left(S_i \prod_{j=1}^{m} P\left(f_j \mid S_i\right)\right),$$

donde f_j representa el vector de características o atributos, mientras que S_i representa el sentido de una palabra en particular. Por lo tanto, el sentido correcto de una palabra será el sentido con el valor de probabilidad condicional más alto. El algoritmo bayesiano, en este caso, demuestra ser ingenuo porque ignora el orden de las palabras; es decir, no logra incorporar realmente las variables del contexto oracional que se utilizan como *input*. A pesar de esto, se trata de un modelo que ha demostrado ser sencillo de implementar y eficiente para el procesamiento de recursos lingüísticos informatizados extensos. No obstante, la calidad de la clasificación estará supeditada a la necesidad de incorporar una mayor cantidad de fuentes de información lingüística. Aun así, el proceso se basa en parámetros exclusivamente estocásticos, y, al realizar la clasificación, la máquina solamente es capaz observar la frecuencia tanto de las palabras objetivo como del contexto oracional.

Diferentes autores (Mooney, 1996; Rish, 2001; Widlak, 2004; Carpuat y Wu, 2005; Eberhardt y Danks, 2011) han justificado empíricamente el desarrollo de sistemas de desambiguación léxica automática utilizando este algoritmo frente a otros disponibles. En efecto, al realizar comparaciones experimentales entre diferentes algoritmos de aprendizaje automático que se utilizan para resolver la desambiguación léxica mediante la clasificación de palabras considerando el contexto oracional (de redes neuronales,

árboles de decisión, basados en reglas, basados en casos, modelo de máxima entropía, *Boosting model*, y *Kernel PCA-based model*), a pesar de su ya mencionada simpleza matemática, el algoritmo bayesiano ingenuo resulta ser significativamente más eficiente para las tareas específicas de desambiguación léxica automática, particularmente en el ámbito de la traducción automática.

Como se mencionó anteriormente, esta técnica está basada en la suposición de independencia condicional: la ocurrencia de una palabra en un texto, dada una clase *x*, es independiente de la ocurrencia de cualquier otra palabra en el mismo texto dada la misma clase *x*. La crítica fundamental a esta suposición proviene desde la lingüística teórica y la ciencia cognitiva pues, en la realidad de las lenguas naturales, la suposición de independencia condicional es incorrecta en tanto las palabras dependen unas de otras y se influencian mutuamente en distintos niveles de análisis lingüístico. No obstante, la misma suposición ha demostrado ser, desde el punto de vista probabilístico, una ventaja.

En resumen, la mayoría de las propuestas de métodos de desambiguación léxica automática basadas en el algoritmo bayesiano ingenuo exponen puntos en común relevantes, también entendidos como ventajas en relación con los métodos basados en métricas. Se pueden resumir en tres puntos:

1) Es un algoritmo llamado sencillo o simplista, dentro la gama de posibilidades para el aprendizaje automático.

2) Es posible incluir un alto número de *rasgos* o características para poder capturar información lingüística que sea necesaria en el proceso de elección de probabilidad; es decir, este método no se limita a la información que provee el cotexto, y puede considerar criterios de análisis provistos por un lingüista.

3) Tiene un desempeño consistentemente sobresaliente, pero que depende de las características del corpus y del número de rasgos.

2.2.2 *Ventajas y desventajas de la utilización del algoritmo Bayesiano Ingenuo para la investigación lingüística*

Dentro de las diferentes técnicas para la desambiguación léxica automática, el aprendizaje automático supervisado ha sido ciertamente una de las más utilizadas en diferentes sistemas que realizan tareas de PLN. Específicamente, según Márquez, Escudero, Martínez y Rigau (2006), la implementación del modelo bayesiano ingenuo se posiciona como un clasificador simple dentro de toda la gama de sistemas disponibles para el aprendizaje automático supervisado (método de los *k*-vecinos más próximos, árboles de decisión, regresión lineal, máquinas de vectores de soporte, entre otros sistemas referenciados en el Capítulo dos). No obstante, una de las ventajas que presenta para la investigación lingüística es su simplicidad y rapidez, además de la posibilidad de incluir un gran número de atributos o características, entendidas a su vez como palabras de contenido para la captura de la información necesaria durante el proceso de clasificación.

La implementación del algoritmo bayesiano ingenuo que se presenta en este capítulo utiliza un número reducido de características correspondientes a palabras que ocurren dentro de una ventana contextual definida, en la que se encuentra la palabra objetivo para el proceso de desambiguación. Así, se trata de un modelo que selecciona un número restringido de palabras para disminuir el número de características utilizadas, con el objetivo de aumentar el rendimiento del proceso de desambiguación. Este método, entonces, se limita a la información proporcionada por el contexto sintáctico y su relación con el corpus de entrenamiento. A partir de lo anterior, se pueden establecer diferentes

críticas, a las que hemos llamado *inadecuaciones*. Estas no solo son relevantes para discutir el modelo bayesiano aplicado a la desambiguación léxica automática en particular, sino también para establecer una crítica a los métodos supervisados en general:

a. Inadecuación epistemológica: Si bien los modelos para el aprendizaje automático han demostrado ser altamente eficientes en la aplicación de diversas tareas de PLN, particularmente en sistemas expertos, se trata de algoritmos cuyos fundamentos evidencian una mínima comprensión del fenómeno del lenguaje humano. En este sentido, el ámbito de la inteligencia artificial tiene por objetivo reproducir los patrones mediante los que funcionan la mente y el lenguaje. Dada esa definición, los métodos estocásticos no logran simular artificialmente la manifestación de una capacidad cognitiva humana, sino que descansan exclusivamente en la eficiencia de los resultados y en la inferencia estadística.

b. Inadecuación explicativa: Los modelos estocásticos en general y bayesiano ingenuo en particular fundamentan su desempeño en los datos de entrenamiento y en la información que provee el contexto oracional. Sin embargo, el análisis de los resultados, sobre todo para un sistema de desambiguación léxica, no permite acceder a ningún mecanismo que explique el comportamiento de los datos, excepto la posibilidad de añadir más información con el objetivo de provocar resultados diferentes. Este problema también es conocido como la opacidad de los algoritmos en inteligencia artificial, y plantea el problema de que el conocimiento de la manera en la que funcionan los modelos estadísticos y su potencial explicativo es limitado, pero sus resultados son altamente eficientes y, por tanto, validados.

c. Inadecuación lingüística: La única variable que puede provocar cambios en los resultados de un sistema de aprendizaje automático supervisado es el volumen de datos. En este sentido, el funcionamiento de cualquier modelo requerirá de una cantidad alta de datos para poder ser ejecutado con resultados que puedan ser válidos. Esta dependencia del volumen del recurso lingüístico informatizado que se utilice, tanto como corpus de entrenamiento como de prueba, puede resultar inviable en muchos casos.

2.3 *Tareas de preprocesamiento*

Para el desarrollo del experimento, se establecieron tres tareas de preprocesamiento junto con sus respectivas tareas en DAMIEN. La primera tarea de preprocesamiento consistió en extraer los datos textuales correspondientes a la ventana contextual para cada una de las instancias seleccionadas. Específicamente, se realizó un procedimiento para la extracción de un archivo *.txt* que almacenara la información del cotexto (Figura 1).

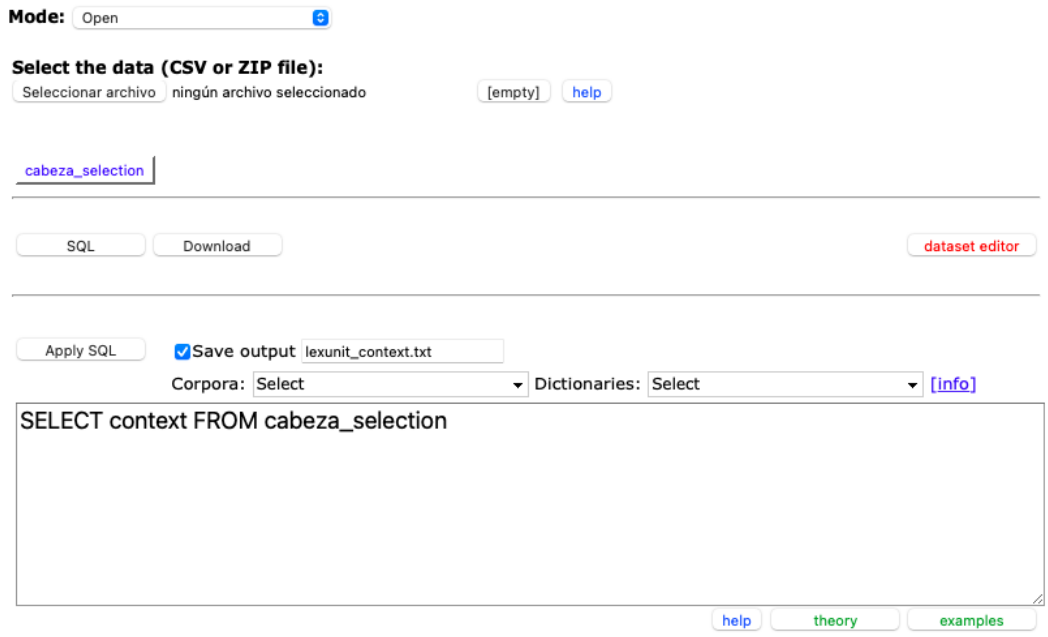

Figura 1: ***Extracción de la ventana contextual en DAMIEN***

La segunda tarea de preprocesamiento consistió en generar una colección de documentos sin anotar. Para esto, la secuencia que se ejecutó en DAMIEN fue el cambio de tamaño (*file resizing*) del archivo *.txt*, mediante la aplicación de una expresión regular *\n* para dividir el contenido de cada cotexto en 120 documentos, como se evidencia en la siguiente Figura 2:

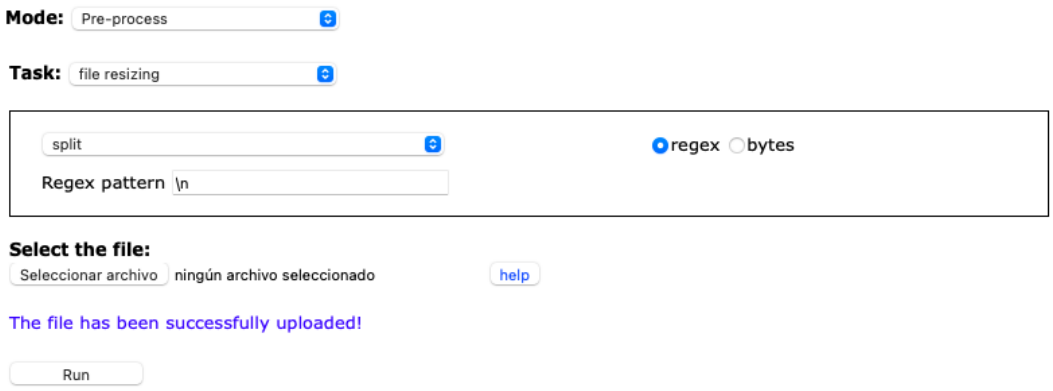

Figura 2: ***Generación de una colección de documentos (sin anotar) en DAMIEN***

La tercera tarea de preprocesamiento fue extraer las etiquetas *senseID*, correspondientes a cada uno de los sentidos seleccionados para las palabras objetivo en la colección de documentos. Al igual que en la primera tarea, este procedimiento se llevó a cabo mediante la ejecución de un comando SQL, como se muestra en la Figura 3:

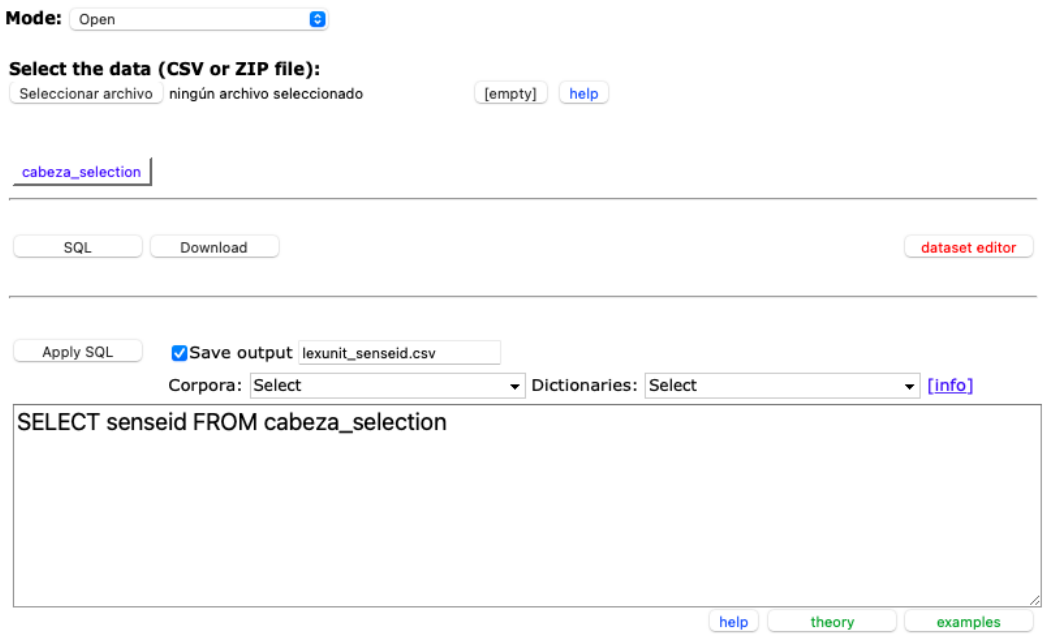

Figura 3: ***Extracción de etiquetas senseID en DAMIEN***

Finalmente, las tres tareas de preprocesamiento anteriores y sus respectivas secuencias en DAMIEN se presentan de manera pormenorizada en la Tabla 1:

Tabla 1: ***Tareas de preprocesamiento para experimentos de aprendizaje automático***

Nº	TAREA	SECUENCIA EN DAMIEN
1	Extracción de la ventana contextual	1. Extraer un archivo *.txt* para la columna *context* 2. Corpus > Open 3. Cargar un *.zip* con la tabla en *.csv* 4. Aplicar comando SQL: SELECT context FROM lexunit_selection (el resultado se genera en *.csv*) 5. Guardar como *.txt* 6. Guardar resultado
2	Generación de una colección de documentos (sin anotar)	1. Cargar un archivo .txt 2. Corpus > Pre-process > File resizing > Split (regex \n) 3. Guardar resultado
3	Extracción de etiquetas *senseID*	1. Extraer un archivo .csv para *senseID* 2. Corpus > Open 3. Comando SQL: SELECT senseID FROM *.csv 4. Guardar resultado

2.4 *Tareas de procesamiento*

Se establecieron cuatro tareas de procesamiento y sus respectivas secuencias en DAMIEN. La primera tarea (cuarta en la secuencia general) consistió en generar la primera matriz *n-grama/documento*. Para la ejecución de esta tarea fue necesario procesar la colección de documentos correspondiente a los cotextos que contuviesen cada palabra objetivo, para así establecer un análisis de la frecuencia de aparición de unidades léxicas en cada una de las instancias en análisis. Las condiciones para la implementación de esta

secuencia fueron: a) que el procesamiento se realice en lengua española; y b) que la captura corresponda a unigramas representados por cada raíz (*stem*)[8] y su frecuencia absoluta. Además, se estableció una lista de palabras vacías o palabras de función (*stopwords*); es decir, no fueron consideradas en la matriz las palabras funcionales, con el objetivo de eliminar el ruido documental. La secuencia en DAMIEN se puede revisar en la Figura 4:

Figura 4: ***Generación de una matriz n-grama/documento en DAMIEN***

La segunda tarea de procesamiento (quinta en la secuencia general), consistió en la generación de una lista de inicio (etiquetada como *startlist*) con el objetivo de filtrar aquellas palabras con mayor peso estadístico dentro del corpus, mediante la aplicación de la medida de información mutua.

La medida de información mutua determina la reducción de la incertidumbre de una variable dado el valor conocido de otra variable; es decir, calcula la cantidad de información que se puede obtener a partir de una variable aleatoria, considerando un valor conocido (Veyrat-Charvillon y Standaert, 2009; Boudiaf, Roni, Masud, Granger, Pedersoli, Piantanida y Ben-Ayed, 2020). Así, en términos estadísticos, esta medida se utiliza para calcular la dependencia entre dos variables aleatorias. Según lo anterior, la medida de información mutua entre dos variables aleatorias e y se puede definir como:

$$I(x_i + y_j) = log \frac{P(x_i|y_j)}{P(x_j)}$$

Específicamente, para esta tarea de procesamiento se midió la cantidad promedio de información que ciertas unidades léxicas transmiten o proyectan sobre otras unidades léxicas presentes en cada colección de documentos. El resultado fue una lista de las palabras que presentaron una mayor significancia estadística luego del análisis de frecuencia proporcionado por la matriz *n-grama/documento*. Finalmente, se utilizó como

[8] En cuanto al concepto de *stem*, según Bauer (2004), se trata de un término que se utiliza para designar una unidad interna siempre presente en un lexema, que permanece estable cuando se han extraído todos los afijos presentes a la vez que incluye los potenciales morfemas flexivos. Así, se ha decidido trabajar con la variable *stem* debido a la frecuencia y relevancia en el corpus de las distintas flexiones de número para cada una de las unidades léxicas en análisis.

filtro el 25% superior de esta lista. La siguiente figura muestra, en la interfaz de DAMIEN, la secuencia antes descrita.

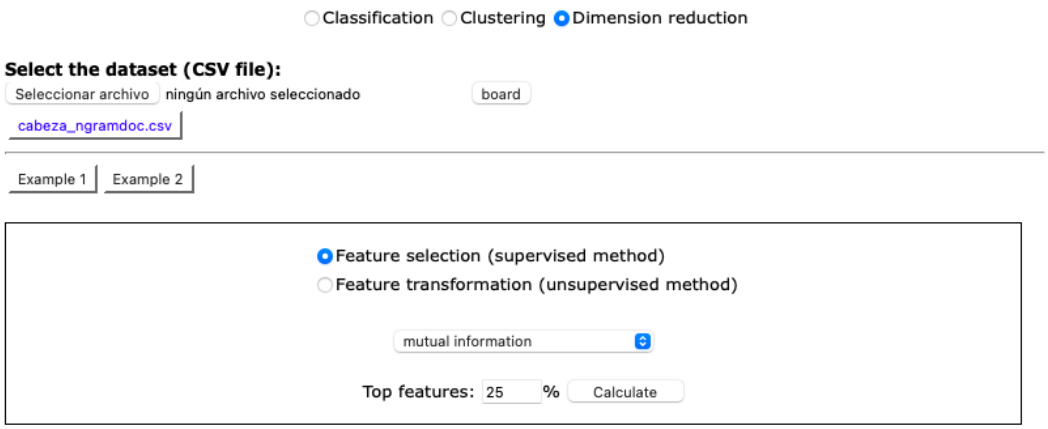

Figura 5: ***Generación de una lista de inicio en DAMIEN***

En la tercera tarea de procesamiento (sexta en la secuencia general) se generó una segunda matriz *n-grama/documento*, considerando el filtro de la *startlist* creada a partir de la aplicación de la medida de información mutua:

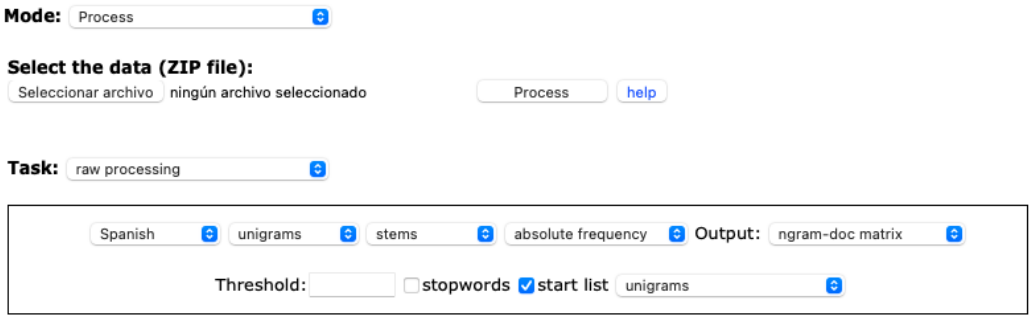

Figura 6: ***Generación de una matriz n-grama/documento con lista de inicio en DAMIEN***

La cuarta tarea de procesamiento, y séptima en la secuencia general, fue fundamental para la ejecución del proceso de desambiguación léxica automática. Consistió en la generación de una matriz *n-grama/documento* filtrada y anotada con los sentidos correspondientes para cada una de las palabras objetivo presentes en los documentos. Para esto, se incluyó en la matriz una nueva columna con los sentidos que fueron extraídos en un paso anterior, mediante la aplicación del comando de unión a la derecha (*join right*), incluyendo el símbolo barra "|" como separador. En la siguiente figura se presenta la interfaz en DAMIEN para esta secuencia:

Figura 7: ***Secuencia join right para generar una matriz filtrada y anotada en DAMIEN***

Finalmente, en la siguiente tabla se presentan las cuatro tareas de procesamiento y sus respectivas secuencias en DAMIEN, que incluyen cada uno de los pasos y configuraciones de manera pormenorizada:

Tabla 2: ***Tareas de procesamiento para experimentos de aprendizaje automático***

Nº	TAREA	SECUENCIA EN DAMIEN
4	Generación de la primera matriz *n-grama/documento*	1. Corpus > Process > Task > Raw processing 2. Settings = Spanish; unigrams; stems; absolute frequency 3. Output = doc-ngram matrix 4. Stopwords [functional] 5. Guardar resultado
5	Generación de una *startlist* para filtrar palabras con mayor peso estadístico en el corpus	1. Mining > Dimension reduction 2. Feature selection = mutual information 3. Top Features = 25 4. Guardar resultado
6	Generación de matriz *n-grama/documento* con *startlist*	1. Aplicar el filtro = lexitem_collection_startlist.zip 2. Corpus > Process > Task > Raw processing Settings = Spanish; unigrams; stems; absolute frequency 3. Output = doc-ngram matrix 4. Startlist = lexunit_startlist 5. Guardar resultado
7	Generación de matriz *n-grama/documento*, filtrada y anotada	1. Aplicar comando JOIN (right) para incluir una columna con los *senseID* de cada documento 2. Guardar resultado

2.5 *Tarea de minería textual*

La tarea de minería textual, correspondiente a la octava tarea en la secuencia general, consiste en la validación cruzada de las 120 instancias procesadas en la matriz *n-grama/documento* filtrada y anotada; es decir, en la división aleatoria de los datos textuales en una cantidad determinada de grupos del mismo tamaño considerando la columna *senseID* como el atributo de clase. En este caso, se generaron aleatoriamente tres *trainings sets* y tres *test sets*, cada uno con 40 instancias. Este procedimiento en DAMIEN se puede ver en la Figura 8:

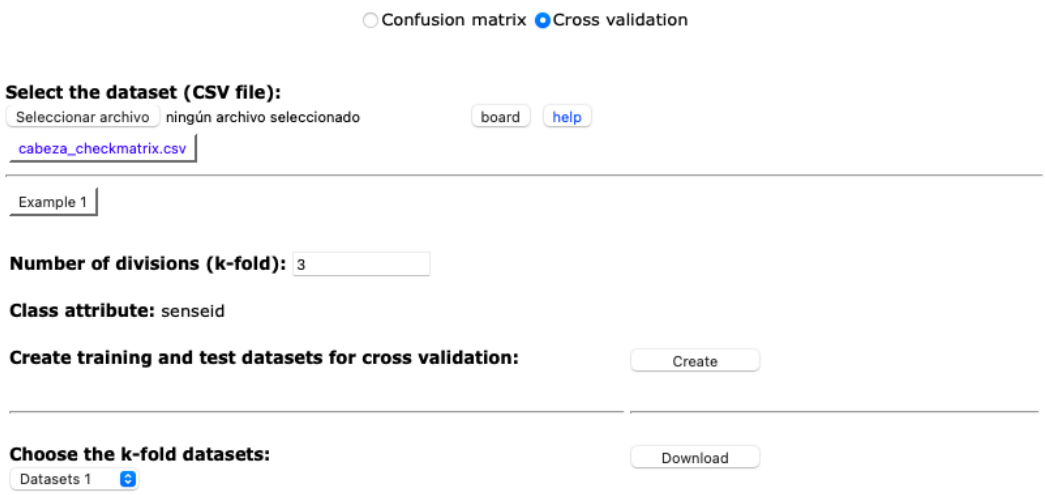

Figura 8: ***Validación cruzada en DAMIEN***

La secuencia pormenorizada en DAMIEN para la tarea de validación cruzada se expone en la Tabla 3:

Tabla 3: ***Tarea de minería textual para experimentos de aprendizaje automático***

Nº	TAREA	SECUENCIA EN DAMIEN
8	Validación cruzada	1. Evaluation > Cross Validation Settings: *k*-fold = 3 2. El resultado corresponde a la creación de tres carpetas con los archivos *training.txt*, *test.txt* y *predicted.txt*.

2.6 *Tareas de evaluación*

Se determinaron dos tareas de evaluación con las que concluyeron los experimentos de aprendizaje automático. La primera tarea (novena en la secuencia general) consistió en la aplicación del algoritmo *Naïve Bayes* para la clasificación de cada *test dataset* basado en el respectivo *training dataset*. Para realizar esta secuencia, se cargaron los conjuntos de datos generados en la validación cruzada. Finalmente, el correspondiente atributo de clase fue la columna *senseID*, identificado automáticamente. En la Figura 9 se puede ver la interfaz de DAMIEN:

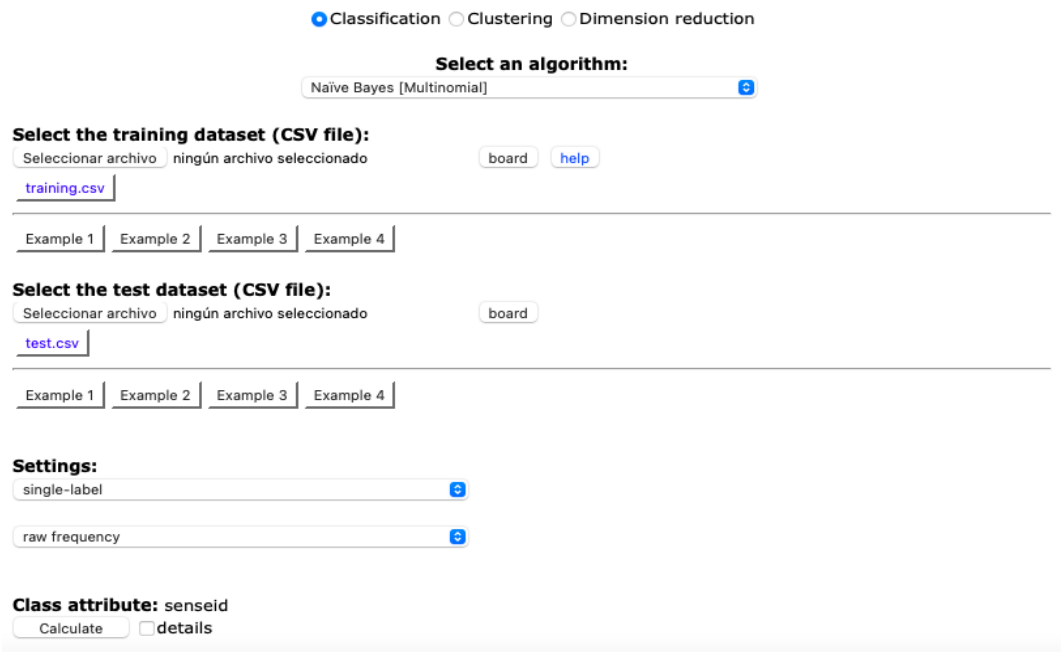

Figura 9: ***Aplicación del algoritmo bayesiano ingenuo en DAMIEN***

La segunda tarea de evaluación, correspondiente a la décima en la secuencia general, es la generación de una matriz de confusión para la evaluación de cada sistema de desambiguación. Este procedimiento requiere crear carpetas con los datos para cada *senseID* por *dataset*. Después, en cada carpeta se deben incluir los correspondientes documentos con los valores tanto esperados como predichos. La interfaz DAMIEN para esta tarea, junto con un ejemplo de matriz de confusión, se exponen en las Figuras 10 y 11:

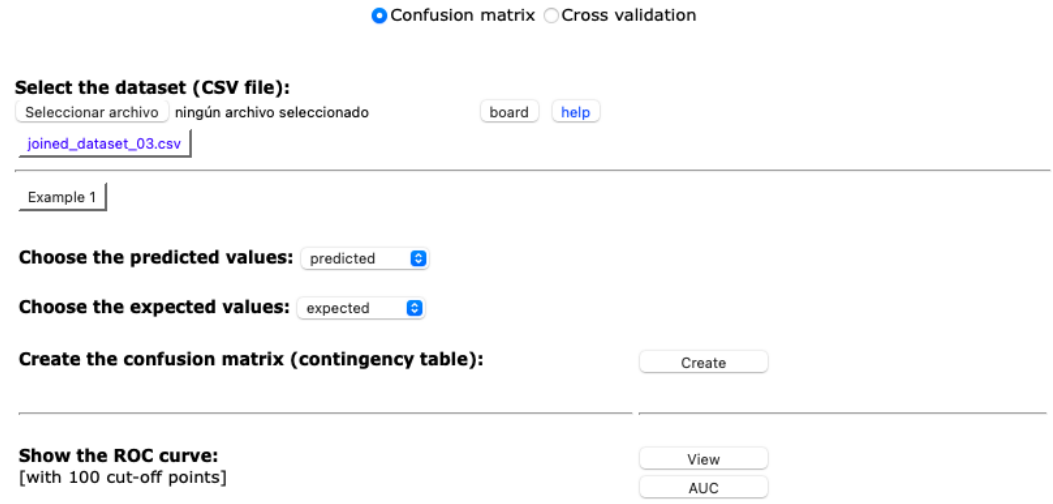

Figura 10: ***Generación de una matriz de confusión en DAMIEN***

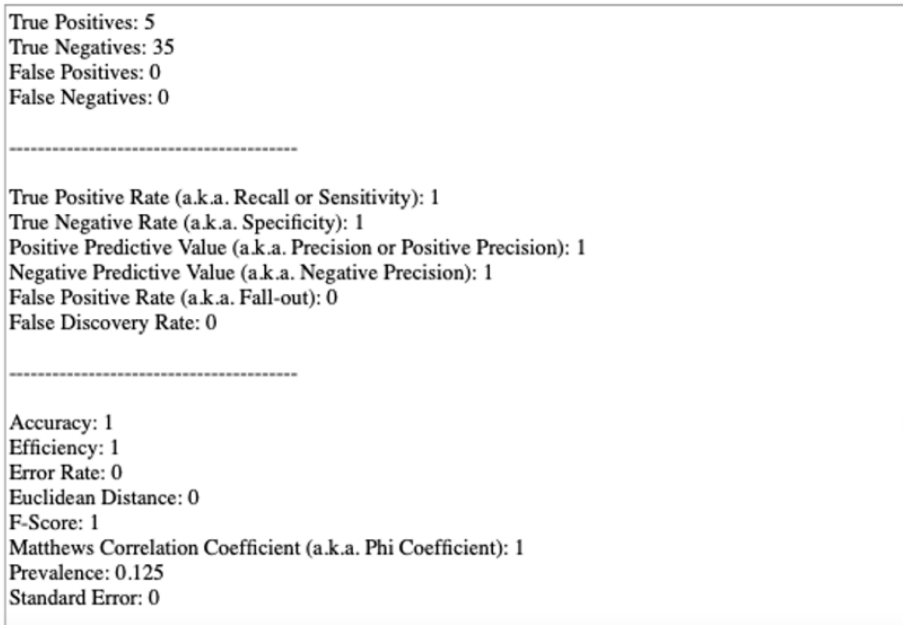

Figura 11: ***Ejemplo de matriz de confusión en DAMIEN***

Finalmente, las tareas de evaluación y su secuencia de pasos en DAMIEN se muestran de manera detallada en la siguiente tabla:

Tabla 4: ***Tareas de evaluación para experimentos de aprendizaje automático***

Nº	TAREA	SECUENCIA EN DAMIEN
9	Aplicación de algoritmo bayesiano ingenuo	1. Mining > Classification > *Naïve Bayes* (multinomial) *2.* Seleccionar y cargar cada *training dataset* en *.csv* 3. Seleccionar y cargar *test dataset* en *.csv* Settings = single-label; raw frequency 4. Class attribute (lo identificará automáticamente desde la tabla) = *senseid* 5. Los resultados se deben guardar como un archivo *predicted.csv* en las carpetas correspondientes para cada *dataset*.
10	Generación de una matriz de confusión para la evaluación del sistema	1. Crear carpetas con los datos para cada *senseID* por *dataset.* En cada carpeta se deben incluir los correspondientes documentos *a_expected.txt* y *b_predicted.txt* 2. Aplicar comando JOIN (*right*) para generar la tabla de evaluación. 3. Reemplazar las etiquetas de s*enseID* por los valores de 0 y 1, correspondientes con cada uno de los sentidos en evaluación. 4. Reemplazar los nombres de las columnas, de izquierda a derecha, por las etiquetas *expected* y *predicted.* 5. Evaluation > Confusion matrix Guardar resultado

2.7 *Sistema de desambiguación léxica automática*

El sistema de desambiguación basado en aprendizaje automático se desarrolló examinando los casos de ambigüedad de las unidades léxicas “cabeza”, “cara” y “carta”, y considerando los sentidos disponibles en la base de conocimiento léxico-conceptual

FunGramKB[9] (Periñán-Pascual y Arcas-Túnez, 2004, 2010; Jiménez-Briones y Luzondo-Oyón, 2011; Periñán-Pascual, 2012). Todos estos sentidos fueron testeados a partir de un corpus de entrenamiento etiquetado manualmente con 120 instancias, que incluyeron cada palabra objetivo junto con una ventana contextual. Este conjunto de instancias, para efectos de la evaluación, fue dividido en tres *datasets* aleatorizados para cada uno de los sentidos en análisis. Luego de la ejecución del algoritmo bayesiano ingenuo, se realizó una evaluación mediante una matriz de confusión para determinar, en última instancia, el macropromedio correspondiente a la media armónica (o puntaje F), que establecería los resultados del sistema mediante una función entre los valores promedio de precisión y cobertura para cada sentido. Su relevancia radica en la posibilidad de representar de manera eficiente la distribución de las clases, para así establecer un puntaje armónico para el desempeño del sistema cuyo valor máximo posible es igual a 1. Lo anterior se formaliza como sigue:

$$Puntaje\ F = 2 \times \frac{precisión\ \times\ cobertura}{precisión\ +\ cobertura}$$

3. RESULTADOS Y DISCUSIÓN

3.1 *Resultados del sistema de desambiguación automática para la unidad léxica "cabeza"*

En el caso de la unidad léxica "cabeza", se consideraron los siguientes sentidos:

1) +HEAD_00: La parte superior o frontal del cuerpo en animales, que contiene la cara y el cerebro.

2) +INTELLIGENCE_00: La habilidad para pensar, sentir e imaginar cosas.

3) +CHIEF_00: Una persona que está a cargo.

4) +LÍDER_00: Una persona que gobierna, guía o inspira a otros.

A continuación, se presentan las tablas con el resumen de la matriz de confusión para cada sentido, junto con los resultados para el sistema de desambiguación léxica automática.

Tabla 5: ***Matriz de confusión para el sentido +HEAD_00 de "cabeza"***

ÍTEM LÉXICO	CABEZA						
SenseID	+HEAD_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F

[9] FunGramKB (*Functional Grammar Knowledge Base*) es una base de conocimiento léxico-conceptual-gramatical multipropósito, cuyo objetivo es el procesamiento de las lenguas naturales. FunGramKB tiene como punto de partida la premisa de que los sistemas complejos para PLN deben cumplir con el objetivo de administrar e interpretar información lingüística. Más información acerca de este proyecto se encuentra disponible en http://www.fungramkb.com/nlp.aspx.

1	14	11	11	4	0,56	0,777	0,651
2	15	8	11	6	0,652	0,714	0,681
3	11	9	13	7	0,55	0,611	0,578
			Promedio		**0,587**	**0,701**	**0,637**

En cuanto al sentido +HEAD_00, en los tres *datasets* se mantuvo un rendimiento homogéneo, considerando específicamente los resultados de la dispersión para el promedio de la media armónica [$\underline{X}_F$ = 0,637; *DE* = 0,05], equivalente a un 64%. Los desempeños particulares se caracterizaron por una mayor precisión [*P* = 0,652] y cobertura [*C* = 0,714], correspondientes a un 68% de rendimiento para el *dataset* dos [*F* = 0,681].

Tabla 7: ***Matriz de confusión para el sentido +CHIEF_00 de "cabeza"***

ÍTEM LEXICO	CABEZA						
SenseID	+CHIEF_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	3	1	29	7	0,75	0,3	0,428
2	1	4	30	6	0,2	0,166	0,181
3	3	3	26	8	0,5	0,272	0,352
			Promedio		**0,483**	**0,246**	**0,32**

En el caso del sentido +CHIEF_00, se observó un resultado bajo [< 50%] para el promedio de la media armónica [$\underline{X}_F$ = 0,32; *DE* = 0,126], equivalente a un 32% de rendimiento. Si bien los *datasets* uno y tres mostraron un desempeño similar, del 43% y 35% respectivamente, el *dataset* dos obtuvo resultados de precisión [*P* = 0,2] y cobertura [*C* = 0,166] particularmente deficientes, equivalentes a un 18%.

Tabla 8: ***Matriz de confusión para el sentido +LEADER_00 de "cabeza"***

ÍTEM LEXICO	CABEZA						
SenseID	+LEADER_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	4	6	26	4	0,4	0,5	0,444
2	1	4	28	7	0,2	0,125	0,153
3	4	7	27	2	0,363	0,666	0,470
			Promedio		**0,321**	**0,430**	**0,356**

Los resultados para el sentido +LEADER_00 corresponden a un desempeño del 36% considerando el promedio para la media armónica $\left[\underline{X}_F = 0{,}356;\ DE = 0{,}175\right]$. Si bien se trata de un resultado bastante homogéneo en cuanto a la dispersión, el *dataset* dos obtuvo un rendimiento bajo para los indicadores de precisión [$P = 0{,}2$] y cobertura [$C = 0{,}125$], con puntaje F equivalente a un 15%.

Tabla 9: ***Matriz de confusión para el sentido +INTELLIGENCE_00 de "cabeza"***

ÍTEM LEXICO	CABEZA						
SenseID	+INTELLIGENCE_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	0	1	35	4	0	0	*NaN*
2	2	5	30	3	0,285	0,4	0,333
3	0	3	32	5	0	0	*NaN*
			Promedio		**0,095**	**0,133**	**0,333**

Los resultados para el sentido +INTELLIGENCE_00 fueron los que arrojaron el rendimiento más irregular del sistema, considerando que el promedio para la media armónica [$\underline{X}_F = 0{,}333$; $DE = 0{,}192$], equivalente a un *33%*, es una expresión válida solamente para los valores de precisión [$P = 0{,}285$] y cobertura [$P = 0{,}4$], correspondientes al *dataset* dos. Lo anterior se justifica dado que, para todas las casillas correspondientes a valores numéricos en los que se indique *NaN* (*not a number*), no ha sido posible establecer un resultado y, por tanto, esa variable no fue considerada en el análisis.

Tabla 10: ***Resultados del sistema de desambiguación automática para "cabeza"***

	M PRECISIÓN	M COBERTURA	M PUNTUAJE F
+HEAD_00	0,587	0,701	0,637
+CHIEF_00	0,483	0,246	0,320
+LEADER_00	0,321	0,430	0,356
+INTELLIGENCE_00	0,095	0,133	0,333
Macropromedio del sistema	**37,15%**	**37,75%**	**41,15%**

El sistema de desambiguación automática para la unidad léxica "cabeza", que considera cuatro sentidos disponibles en la base de conocimiento, logra un rendimiento promedio del 41,15% [$DE = 0{,}151$], a partir de cuya dispersión se puede establecer un desempeño homogéneo, con la excepción del sentido +HEAD_00 que presenta un promedio de la media armónica [$F = 0{,}637$] superior en 0,3 puntos al puntaje *F* más alto para el resto de los sentidos, correspondiente a +LEADER_00 [$F = 0{,}356$]. Estos resultados, además, indican que la cantidad de sentidos fue en desmedro de la capacidad del sistema para clasificar correctamente las instancias en análisis. Según lo anterior, el rendimiento más bajo fue el sentido +CHIEF_00, con un *32%* [$F = 0{,}32$].

3.2 *Resultados del sistema de desambiguación automática para la unidad léxica "cara"*

Para la unidad léxica "cara" consideraron los siguientes sentidos:

1) +FACE_00: La parte delantera de la cabeza desde la frente hasta el mentón y de oreja a oreja.

2) +SIDE_00: Una superficie que forma parte del exterior de un objeto.

Los resultados para el sistema de desambiguación automática son los siguientes:

Tabla 11: ***Matriz de confusión para el sentido +FACE_00 de "cara"***

ÍTEM LÉXICO	CARA						
SenseID	+FACE_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	16	6	3	15	0,727	0,516	0,603
2	23	5	2	10	0,821	0,696	0,754
3	24	9	3	4	0,727	0,857	0,786
			Promedio		**0,758**	**0,690**	**0,714**

Los resultados para el sentido +FACE_00 se caracterizan por un alto desempeño del sistema, correspondiente a un 71% de rendimiento promedio según la media armónica, con una dispersión homogénea [$\underline{X}_F = 0{,}714$; $DE = 0{,}097$].

Específicamente, el *dataset* dos alcanzó el valor más alto de precisión [$P = 0{,}821$], mientras que el *dataset* tres obtuvo el valor más alto de cobertura [$C = 0{,}857$]. Estos resultados representaron, en términos generales, el desempeño más alto para los sistemas de desambiguación propuestos.

Tabla 12: ***Matriz de confusión para el sentido +SIDE_00 de "cara"***

ÍTEM LÉXICO	CARA						
SenseID	+SIDE_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	3	15	16	6	0,166	0,333	0,222
2	2	10	23	5	0,166	0,285	0,210
3	3	4	24	9	0,428	0,25	0,315
			Promedio		**0,253**	**0,289**	**0,249**

En cuanto al sentido +SIDE_00, el rendimiento promedio equivale a un 25% para la media armónica [$\underline{X}_F = 0{,}249$; $DE = 0{,}05$]. Si bien este resultado es bajo, el *dataset* tres mostró una precisión más alta [$P = 0{,}428$], aunque aún < 50%. No obstante, la dispersión promedio se muestra homogénea.

Tabla 13: ***Resultados del sistema de desambiguación automática para "cara"***

	M PRECISIÓN	M COBERTURA	M PUNTUAJE F
+FACE_00	0,758	0,690	0,714
+SIDE_00	0,253	0,289	0,249
Macropromedio del sistema	**50,55%**	**48,95%**	**48,15%**

El sistema de desambiguación automática para la unidad léxica "cara" obtuvo un rendimiento promedio del 48,15% [$\underline{X}_F = 0{,}481$; $DE = 0{,}328$]. No obstante, hubo una diferencia de *0,465* puntos en el desempeño de los sentidos en análisis. Por tanto, se establece que el sistema no logra realizar la tarea de clasificación de manera eficiente, sobre todo para el caso de +SIDE_00.

3.3 *Resultados del sistema de desambiguación automática para la unidad léxica "carta"*

Para el análisis de la unidad léxica "carta", se consideraron los siguientes sentidos:

1) +LETTER_00: Un mensaje escrito dirigido a una persona u organización.

2) +CARD_00: Un pequeño trozo de papel grueso y rígido con números o imágenes, que se usa para jugar un juego en particular.

3) $MENU_00: Una lista de platos disponibles en un restaurante.

A continuación, se presentan las tablas con el resumen de los resultados para el sistema de desambiguación automática.

Tabla 14: ***Matriz de confusión para el sentido +LETTER_00 de "carta"***

ÍTEM LÉXICO	CARTA						
SenseID	+LETTER_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	14	6	11	9	0,7	0,608	0,651
2	17	7	11	5	0,708	0,772	0,739
3	20	4	10	6	0,833	0,769	0,8
			Promedio		**0,747**	**0,716**	**0,730**

El sentido +LETTER_00 obtiene un promedio para la media armónica correspondiente a un 73% de rendimiento [$\underline{X}_F = 0{,}730$; $DE = 0{,}074$]. Estos resultados muestran un desempeño alto y homogéneo según la dispersión de los datos. En cuanto a los resultados particulares, la precisión más alta la alcanzó el *dataset* tres [$P = 0{,}833$], mientras que la cobertura más alta correspondió al *dataset* dos [$C = 0{,}772$].

Tabla 15: ***Matriz de confusión para el sentido +CARD_00 de "carta"***

ÍTEM LÉXICO	CARTA						
SenseID	+CARD_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	5	9	23	3	0,357	0,625	0,454
2	4	6	27	3	0,4	0,571	0,470
3	2	4	29	5	0,333	0,285	0,307
			Promedio		**0,363**	**0,494**	**0,410**

En cuanto al sentido +CARD_00, el promedio para la media armónica equivale a un 41% de rendimiento [$\underline{X}_F = 0{,}410$; $DE = 0{,}089$]. En términos generales, los resultados indican una dispersión homogénea, lo que se traduce en un desempeño consistentemente < 50%. El rendimiento más alto lo alcanzó el *dataset* dos, tanto para el valor de precisión [$P = 0{,}4$] como de cobertura [$C = 0{,}571$], con un puntaje *F* equivalente al 47%.

Tabla 16: ***Matriz de confusión para el sentido $MENU _00 de "carta"***

ÍTEM LÉXICO	CARTA						
SenseID	$MENU_00						
Dataset	VP	FP	VN	FN	Precisión	Cobertura	Puntaje F
1	4	2	29	5	0,666	0,444	0,533
2	6	0	29	5	1	0,545	0,705
3	5	5	28	2	0,5	0,714	0,588
			Promedio		**0,722**	**0,568**	**0,609**

El sentido $MENU_00 presentó un promedio para la media armónica equivalente a un 61% de rendimiento [$\underline{X}_F = 0{,}609$; $DE = 0{,}087$]. La precisión del *dataset* dos alcanzó el puntaje máximo [$P = 1$], lo que indica que el desempeño de las predicciones correctas fue de un 100%. Se trata de resultados eficientes, pero que evidencian una dispersión más alta que otros sentidos, considerando las diferencias entre los resultados para la media armónica de los tres *datasets*.

Tabla 17: ***Resultados del sistema de desambiguación automática para "carta"***

	PRECISIÓN	COBERTURA	PUNTUAJE F
+LETTER_00	0,747	0,716	0,730
+CARD_00	0,363	0,494	0,410
$MENU_00	0,722	0,568	0,609
Macropromedio del sistema	**61,07%**	**59,27%**	**58,3%**

3.4 *Resultados generales del sistema de desambiguación automática*

El sistema de desambiguación automática para la unidad léxica "carta" alcanzó un desempeño promedio del 58,3% para su media armónica [$\underline{X}_F = 0{,}583$; $DE = 0{,}161$]. Se trata de un sistema de clasificación eficiente en términos generales, pero que muestra una dispersión más alta que los sistemas de "cabeza" y "cara." Eso se debe al impacto que tienen sobre el macropromedio los valores bajos para el sentido +CARD_00, por un lado, y el alto desempeño para el sentido $MENU_00, por otro. Finalmente, los resultados comparados para los macropromedios correspondientes a los sistemas de desambiguación automática aplicando el algoritmo bayesiano ingenuo son los siguientes:

Tabla 18: ***Macropromedios para el sistema de desambiguación automática***

SISTEMA	MACROPROMEDIOS		
	Precisión	Cobertura	Puntaje F
Cabeza	37,15%	37,75%	41,15%
Cara	50,55%	48,95%	48,15%
Carta	61,07%	59,27%	58,3%

En la Tabla 18 se observa que dos de los tres sistemas alcanzan un desempeño < 50% para el promedio de la media armónica. En el caso de "cabeza", con un 41,15% de rendimiento para el promedio del puntaje *F*, si bien hubo una dispersión baja en los resultados pormenorizados, existiría cierta proporcionalidad entre los valores para los errores en la clasificación y la alta cantidad de sentidos disponibles; a saber, cuatro. En el caso de "cara", si bien se establecieron dos sentidos disponibles, los resultados del *dataset* dos fueron tan deficientes que impactaron en el puntaje *F* obtenido como macropromedio. En el caso de "carta", por el contrario, se evidencia un desempeño > 50% en el promedio para la media armónica, lo que indica una proporción eficiente para el número de aciertos en la clasificación.

4. CONCLUSIONES

Dentro de las diferentes técnicas para la desambiguación léxica automática, el aprendizaje automático supervisado ha sido ciertamente una de las más utilizadas en diferentes sistemas que realizan tareas de PLN. En particular, el modelo bayesiano ingenuo se posiciona como un clasificador simple dentro de toda la gama de sistemas disponibles actualmente para el aprendizaje automático supervisado aplicado al PLN, como son el método de los *k*-vecinos más próximos, los árboles de decisión, la regresión lineal, las máquinas de vectores de soporte, la similitud y relación semánticas, entre otros. De este modo, las ventajas que el algoritmo bayesiano ingenuo presenta para la investigación lingüística son su simplicidad y rapidez, además de la posibilidad de incluir un gran número de atributos o características, entendidas a su vez como palabras de contenido para la captura de la información necesaria durante el proceso de clasificación.

El entorno de trabajo DAMIEN, cuyo potencial hemos presentado, ejecuta y resuelve adecuadamente las tareas de procesamiento textual para la desambiguación léxica automática basada en un método de aprendizaje automático. Se trata de una herramienta que permite la realización integrada de las tareas necesarias para la investigación lingüística basada en el análisis de datos textuales, de tal manera que se maximiza tanto el acceso como la eficiencia de herramientas especializadas. En cuanto a nuestra propuesta específica de implementación, el algoritmo bayesiano ingenuo logra seleccionar un número restringido de palabras para disminuir el número de características utilizadas, con el objetivo de aumentar el rendimiento del proceso de desambiguación.

Finalmente, en este trabajo hemos llevado a cabo un experimento para mostrar el procedimiento de desambiguación de tres unidades léxicas con DAMIEN utilizando el método de aprendizaje automático supervisado basado en el algoritmo bayesiano ingenuo, que sienta las bases de la metodología para futuros trabajos en el ámbito del tratamiento de recursos lingüísticos informatizados.

REFERENCIAS

Allen, J. (1995). *Natural Language Understanding*. Redwood City: The Benjamin Cummings Publishing Company.

Aung, N. T., Soe, K. y Thein, N. (2011). A word sense disambiguation system using naïve Bayesian algorithm for Myanmar language. *International Journal of Scientific & Engineering Research*, 9, 1-7.

Bauer, L. (2004). *English Word-formation*. Cambridge: Cambridge Textbooks in Linguistics.

Boudiaf, M., Roni, J., Masud, I., Granger, E., Pedersoli, M., Piantanida, P. y Ben-Ayed, I. (2020). A unifying mutual information view of metric learning: cross-entropy vs. pairwise losses. En A. Vedaldi, H. Bischof, T. Brox y J. M. Frahm (Eds.), *Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science, vol. 12351* (pp. 548-564). Springer: Cham.

Cantos-Gómez, P. (1996). *Lexical Ambiguity, Dictionaries and Corpora*. Murcia: Servicio de Publicaciones, Universidad de Murcia.

Carpuat, M. y Wu, D. (2005). Evaluating the Word Sense Disambiguation Performance of Statistical Machine Translation. En n.a. (Eds.), *Actas del Second International Joint Conference on Natural Language Processing (IJCNLP)* (pp. 120-125). Jefu, Korea: Asian Federation of Natural Language Processing. Sacado de https://aclanthology.org/I05-2021.pdf

Choi, R., Coyner, A., Kalpathy-Cramer, J., Chiang, M. y Campbell, P. (2020). Introduction to machine learning, neural networks, and deep learning. *Translational Vision Science & Technology*, 9(2), 1-12. doi: 10.1167/tvst.9.2.14

Eberhardt, F. y Danks, D. (2011). Confirmation in the cognitive sciences: the problematic case of Bayesian models. *Minds and Machines*, *21*(3), 389-410. doi: 10.1007/s11023-011-9241-3

Escudero, G., Màrquez, L. y Rigau, G. (2000). A comparison between supervised learning algorithms for Word Sense Disambiguation. En n.a. (Eds.), *Actas del Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop* (pp. 31-36). doi: 10.3115/1117601.1117609 Sacado de https://www.cs.upc.edu/~escudero/wsd/00-conll.pdf

Espunya i Prat, A. (1994). Computational linguistics: a brief introduction. *Links & Letters*, 1, 9-23.

Fulmari, A., y Chandak, M. (2014). An approach for Word Sense Disambiguation using modified naïve bayes classifier. *International Journal of Innovative Research in Computer and Communication Engineering Organization* 2(4), 3867-3870.

Gale, W., Church, K. y Yarowsky, D. (1992). A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26, 415-439.

Gamallo, P., Sotelo, S. y Pichel, J. (2014). *Comparing ranking-based and naive bayes approaches to language detection on tweets*. Artículo presentado en el Workshop TweetLID: Twitter Language Identification Workshop at SEPLN 2014. Girona, España. 16 de septiembre.

Gosal, G. (2015). A naïve bayes approach for Word Sense Disambiguation. *International Journal of Advanced Research in Computer Science and Software Engineering* 5(7), 336-340.

Hastie, T., Tibshirani, R. y Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference and Prediction* (2ª ed.). Nueva York: Springer.

James, G., Witten, D., Hastie, T. y Tibshirani, R. (2013). *An Introduction to Statistical Learning with Applications in R*. Nueva York: Springer.

Jiménez Briones, R. y Luzondo-Oyón, A. (2011). Building ontological meaning in a lexico-conceptual knowledge base. *Onomázein*, 23, 11-40.

Jurafsky, D. y Martin, J. (1998). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Nueva Jersey: Prentice Hall.

Li, Y. y Yang, T. (2018). Word embedding for understanding natural language: a survey. En S. Srinivasan (Ed.), *Guide to Big Data Applications. Studies in Big Data*, vol 26. (pp. 83-106). Cham: Springer.

Manning, C. y Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge: The MIT Press.

Márquez, L., Escudero, G., Martínez, D. y Rigau, G. (2006). Supervised corpus-based methods for WSD. En E. Agirre y P. Edmonds (Eds.), *Word Sense Disambiguation: Algorithms and Applications* (pp. 167-216). Cham: Springer.

Mooney, R. (1996). Comparative experiments on disambiguating word senses: an illustration of the role of bias in machine learning and bias learning to disambiguate word senses. En E. Brill y K. Church (Eds.), *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-96)* (pp. 82-91). Pennsylvania: Universidad de Pennsylvania.

Moor, J. (2006). The Dartmouth College Artificial Intelligence conference: the next fifty years. *AI Magazine*, 27(4), 87-91. doi: 10.1609/aimag.v27i4.1911

Núñez-Torres, F. (2013). La representación léxica en el modelo del Lexicón Generativo de James Pustejovsky, *Onomázein, Revista de Lingüística, Filología y Traducción de la Pontificia Universidad Católica de Chile*, 28, 337-345. doi: 10.7764/onomazein.28.9

Periñán-Pascual, C. (2012). En defensa del procesamiento del lenguaje natural fundamentado en la lingüística teórica. *Onomázein, Revista de Lingüística, Filología y Traducción de la Pontificia Universidad Católica de Chile*, 26, 13-48. doi: 10.7764/onomazein.26.01

Periñán-Pascual, C. (2017). Bridging the gap within text-data analytics: a computer environment for data analysis in linguistic research, *Revista de Lenguas para Fines Específicos*, 23(2), 111-132. doi: 10.20420/rlfe.2017.175

Periñán-Pascual, C. y Arcas-Túnez, F. (2004). Meaning postulates in a lexico-conceptual knowledge base. Artículo presentado en *The 15th International Workshop on Databases and Expert Systems Applications.* Recuperado de http://www.fungramkb.com/resources/papers/001.pdf

Periñán-Pascual, C. y Arcas-Túnez, F. (2010). The architecture of FunGramKB. En N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner y D. Tapias (Ed.s), *Proceedings of the Seventh International Conference on Language Resources and Evaluation, European Language Resources Association (ELRA)* (pp. 2667-2674). Valletta, Malta: European Language Resources Association (ERLA).

Raganato, A., Camacho-Collados, J. y Navigli, R. (2017). Word Sense Disambiguation: a unified evaluation framework and empirical comparison. En A. Raganato, J. Camacho-Collados y R. Navigli (Eds.), *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, (pp. 99-110). Valencia: Association for Computational Linguistics.

Rish, I. (2001). An empirical study of the naive bayes classifier. *The IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, 3(22), 41-46.

Veyrat-Charvillon, N. y Standaert, F. (2009). Mutual Information Analysis: How, When and Why? En C. Clavier y K. Gaj (Eds.), *Cryptographic Hardware and Embedded Systems - CHES 2009. CHES 2009. Lecture Notes in Computer Science*, *vol. 5747* (pp. 429-443). Berlin: Springer.

Widlak, M. (2004). *Influence of Word Sense Disambiguation on Text Classification* (Trabajo fin de máster). Universidad de Ottawa, Canadá.