Extracción de palabras clave en documentos no estructurados utilizando Spacy

M. Tovar Vidal[1], E. Santos Rodríguez[1], M. Contreras González[1]
[1] Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla,
Av. San Claudio 14 sur, Ciudad, Universitaria, 72592 Puebla, Pue., México
***Área de participación**: Sistemas Computacionales*

Resumen

Las palabras clave juegan un papel de suma importancia ya que nos permiten caracterizar el contenido de un texto de una forma breve. Debido a ello, la extracción de dichas palabras es un problema competente en distintas áreas de conocimiento como la Recuperación de Información, el Procesamiento de Lenguaje Natural, entre otras. Esta investigación presenta una propuesta de solución para la extracción e identificación de palabras clave a través de un analizador de dependencias y un etiquetado gramatical en documentos electrónicos relacionados con temas de salud escritos en español. Los resultados sugieren que el enfoque presentado puede ser útil para lograr la tarea mencionada anteriormente.

***Palabras clave**: extracción de palabras clave, análisis de dependencias, procesamiento de lenguaje natural, documentos electrónicos.*

Abstract

Keywords play a very important role because they allow us to characterize the content of a text in a brief way. Due to this, the extraction of these words is a competent problem in different areas of knowledge such as Information Retrieval, Natural Language Processing, among others. This research presents a solution proposal for the extraction and identification of key phrases through a dependency analyzer and POS tagging in electronic documents related to health topics written in Spanish. The results suggest that the approach presented may be useful to achieve the task mentioned above.

***Key words**: keywords extraction, dependency parsing, natural language processing, e-documents.*

Introducción

El Procesamiento de Lenguaje Natural (PLN) es el enfoque computacional para analizar texto que está basado tanto en un conjunto de teorías como en un conjunto de tecnologías. Asimismo, puede definirse como una gama teóricamente motivada de técnicas para analizar y representar textos en lenguaje natural en uno o más niveles de análisis lingüístico con el propósito de lograr, de forma humana, el procesamiento de lenguaje para una variedad de tareas o aplicaciones, ya que provee tanto teoría como implementaciones para un rango de aplicaciones, especialmente cualquier aplicación que utilice texto como Recuperación de Información (RI), Extracción de Información (EI), traducción automática, búsqueda de respuestas, sistemas de diálogo, etc., [Liddy, 2001]. Entre estas áreas de aplicación, se encuentra la extracción de palabras clave, que puede definirse como una tarea que identifica automáticamente un conjunto de términos que mejor describen el tema de un documento [Beliga, 2014]. La extracción de palabras clave puede ser realizada manualmente o automáticamente, pero la primera forma de hacerlo consume mucho tiempo y es costoso. De este modo, existe la necesidad de un proceso que extraiga dichas palabras clave de manera automática [Siddiqi y Sharan, 2015]. Las palabras clave son ampliamente usadas para búsquedas en sistemas de Recuperación de Información ya que son fáciles de definir, revisar, recordar y compartir. Las palabras clave extraídas pueden ser utilizadas para construir índices automáticos para una colección de documentos o ser usadas para la representación de un documento en tareas de categorización o clasificación. Entre las distintas aplicaciones de PLN o RI, cuyo núcleo es un resumen extractivo, se encuentran las siguientes: indexación automática, gestión de documentos, descripción semántica de alto nivel, categorización o agrupamiento de texto, documentos o sitios web, recuperación de categoría

rec. 31/ago/2018, acc. 14/sep/2018

cruzada, construcción de diccionarios de dominio específico, reconocimiento de entidades nombradas, detección de tópicos, etc., [Beliga, 2014].

En el presente trabajo se describe una propuesta para la extracción de palabras clave en documentos electrónicos sobre salud en español mediante un analizador de dependencias y un etiquetador gramatical. El método consiste en utilizar el árbol de dependencias, además de apoyarse en el etiquetado gramatical, para analizar la estructura de las oraciones y las relaciones que mantienen entre ellas para obtener los sustantivos, verbos o adjetivos (los cuales serán considerados como candidatos para ser palabras clave) de las oraciones presentes en los documentos que conforman el corpus. Luego de hallar las palabras clave candidatas, se identifican a través de un índice y dos números que representan dónde inicia y termina cada una de éstas; posteriormente, se genera un archivo con está información. Finalmente, se evalúa la eficacia del algoritmo propuesto para identificar palabras clave, a través de las medidas de precisión, exhaustividad y la medida estándar F_1.

El documento está estructurado de la siguiente manera: se inicia con la descripción de algunos trabajos relacionados con la extracción de palabras clave. Posteriormente, se presenta la metodología de solución, los resultados obtenidos y finalmente las conclusiones.

Trabajos relacionados

Existen distintas propuestas para la extracción de términos candidatos y palabras claves en documentos científicos. Por lo tanto, a continuación, se muestran diversas soluciones para el problema mencionado anteriormente:

Stauffer y col., [2018] proponen un enfoque para la detección de palabras clave, basado en plantillas, en documentos manuscritos históricos. Dicho enfoque utiliza distintas representaciones gráficas para imágenes de palabras segmentadas y un método de correspondencia. Por otra parte, el desempeño de este sistema está al nivel e incluso supera a otros métodos basados en plantillas o aprendizaje.

SwiftRank es un enfoque estadístico estocástico no supervisado para clasificar palabras clave e identificar oraciones principales dentro de un sólo documento para la extracción genérica de resúmenes. Este método percibe la información destacada de una unidad de texto, que está relacionada con su título correspondiente y la influencia de ésta dependiendo de la posición de la oración en el texto [Lynn y col.,2018].

Benny y Mintu [2015] presentan un método que recopila tweets utilizando una palabra clave específica y, luego, los resume para hallar temas relacionados con la palabra clave. La detección de temas se realiza mediante grupos de patrones frecuentes. Además, se plantean dos algoritmos, TDA (*Topic Detection*) y TCTR (*Topic Clustering and Tweet Retrieval*), que permiten superar el problema de correlación errónea de patrones.

Gonenc y Ylias [2007] abordan el problema de la extracción automática de palabras clave en documentos como una tarea de aprendizaje supervisado. Se propone y describe una técnica de extracción que utiliza cadenas léxicas (un conjunto de palabras relacionadas semánticamente). Asimismo, se sugiere que los resultados son favorables.

En este artículo se propone la extracción de palabras clave a través de la identificación de sustantivos, verbos y adjetivos obtenidos a partir del grafo de dependencias sintácticas de documentos no estructurados del área de la salud.

Metodología

Iniciamos con la propuesta del algoritmo para la extracción de palabras clave y posteriormente se presenta un ejemplo de su uso.

Algoritmo

El siguiente algoritmo fue implementado en el lenguaje de programación Python mediante la utilización de la biblioteca Spacy para Procesamiento de Lenguaje Natural [Honnibal y Montani, 2017]. A continuación, se muestra el pseudocódigo de la solución propuesta para la extracción de palabras clave en documentos sobre temas de salud.

Algoritmo 1: Extracción de palabras clave
Entrada: Un documento en texto plano.
Salida: Un archivo que contendrá un índice de la palabra clave, dónde comienza y dónde termina.

```
Función creaArchivo(index: int, tok_idx:int, tok_leng:int, nom_arch: string):void
 Var: file1:archivo
 file1=Abrir archivo nom_arch en modo anexar
 Escribir index, tok_idx, tok_leng en file1
 Cerrar file1
Fin_Función

Var:index:int, file:archivo, nom_arch:string
Inicio
nlp=spacy.load('es') //Llamada a Spacy con los modelos en español
index=1
file=Abrir archivo nom_arch en modo lectura
doc=nlp(file) //Función de Spacy
Para oracion en doc.sents
 Para tok en sent:
   Si tok.dep_=="amod" y tok.head.dep_=="obj"
    crearArchivo(index, tok.head.idx, tok.head.idx+len(tok.head.text+" "+tok.text, "output_A_*.txt")) //(*) representa
el nombre del  tema del archivo
    index=index+1
   Fin_si
   Si tok.pos_=="NOUN" o tok.pos_=="PROPN" o tok.dep_=="nsubj" y tok.head.pos_=="VERB"
    crearArchivo( index, tok.idx, tok.idx+len(tok.text, "output_A_(*).txt"))
    index=index+1
   Fin_si
   Si tok.dep_=="nsubj" y tok.head.pos_=="VERB"
    crearArchivo(index, tok.head,idx, tok.head.idx+len(tok.head.text), "output_A_(*).txt")
    index=index+1
   Fin_si
  Fin_para
Fin_para
Fin
```

Ejemplo

El algoritmo descrito anteriormente se aplicó a la oración *"El asma afecta las vías respiratorias*" y se utilizó una función de la biblioteca Spacy para mostrar la imagen del árbol de dependencia generado para la oración mencionada, como puede verse en la Figura 1. El algoritmo comienza cargando el texto en la variable doc y posteriormente comienza a buscar, de forma secuencial, en las oraciones del texto las relaciones que poseen cada uno de los tokens con respecto a otros para encontrar los sustantivos. En este caso, se recuperan las palabras *asma y vías* a través de la verificación, mediante el árbol de dependencias, de que el token tenga una dependencia de tipo sujeto nominal (*nsubj*), se comprueba que las etiquetas gramaticales para dicho token sean sustantivos o sustantivos propios (*NOUN* y *PROPN*, respectivamente) y se revisa que la etiqueta gramatical de la cabeza asociada a este token sea un verbo (*VERB*). *Afecta* se obtiene verificando que la dependencia del token es de tipo sujeto nominal y que la etiqueta gramatical del mismo es un verbo. Finalmente, se encuentra *vías respiratorias* revisando que la dependencia del token sea un modificador adjetival (*amod*) y que la dependencia del token cabeza de éste sea de tipo objeto (*obj*). Además, se indica dónde comienza y termina cada una de ellas;

luego se escriben estos datos en un archivo que servirá para su futura evaluación. Los resultados indican que asma, vías y vías respiratorias son palabras claves.

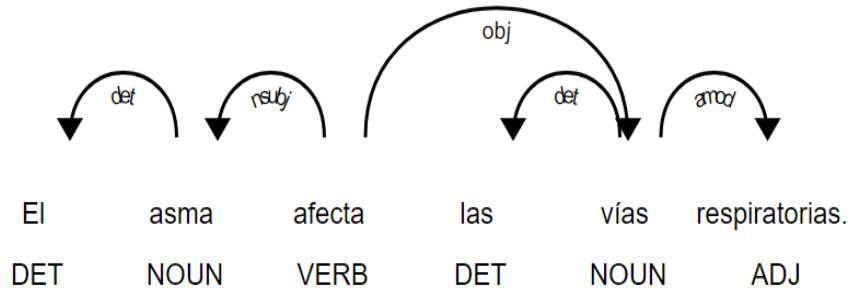

Figura 1. Árbol de dependencia y etiquetado gramatical para una oración en español.

Resultados y discusión

En esta sección se describe la biblioteca Spacy y el corpus empleado en los experimentos, así como las métricas que se utilizarán en la evaluación de la eficiencia del algoritmo. Posteriormente, se presentan los resultados obtenidos.

Spacy

Spacy es una biblioteca para el Procesamiento de Lenguaje Natural, escrita en Python y Cython, que presenta nuevos modelos neuronales para etiquetado, análisis y reconocimiento de entidades. Estos han sido diseñados e implementados desde cero específicamente para Spacy, lo cual permite ofrecer un balance de velocidad, tamaño y precisión incomparables. Los modelos de Spacy son 10 veces más pequeños, 20% más precisos e incluso más baratos de ejecutar [Honnibal y Montani, 2017].

Descripción del corpora

Para realizar la subtarea A de la tarea 3, *eHealt Knowledge Discovery*, propuesto en TASS (Taller de Análisis Semántico de la SEPLN) se utilizaron los documentos del conjunto de datos de entrenamiento, que está conformado por 6 documentos (asuntos sociales y familiares, riñones y sistema urinario, piel, cabello y uñas, pruebas de diagnóstico, asuntos de seguridad y asuntos personales de salud).

Los archivos contienen diversas entradas relacionadas a temas de salud y medicina y han sido procesados para remover todas las etiquetas XML para extraer el contenido textual, en el cual sólo han sido considerados los ítems en español. Una vez limpiados, cada ítem individual fue convertido y se le aplicó post procesamiento para remover cabeceras, notas al pie y elementos similares para convertir las listas HTML en oraciones. Los documentos finales fueron etiquetados manualmente usando Brat con un grupo de anotadores. Después de etiquetar, se aplicó post procesamiento a los archivos de salida de Brat (en formato ANN) para obtener los archivos de salida descritos [Gutiérrez Vázquez, 2018].

Análisis del corpus

Se aplicó el algoritmo a los 6 textos que conforman el corpus y se obtuvieron los resultados mostrados en la tabla 1. En la tabla se muestran los resultados para cada archivo, el cual tiene 3 atributos: tipos, tokens y diversidad léxica. Los tokens son las secuencias de caracteres, los tipos se consideran como las palabras diferentes (vocabulario) y la diversidad léxica es la cantidad de veces, en promedio, que se utiliza una palabra [Bird y col., 2009].

Tabla 1. Tipos (vocabulario), tokens y diversidad léxica de los archivos que conforman el corpus.

Documento	Tipos	Tokens	Diversidad léxica
Asuntos sociales y familiares	418	1056	2.5263
Riñones y sistema urinario	679	2780	4.0942
Piel, cabello y uñas	998	3642	3.6492
Asuntos personales de salud	111	178	1.6036
Pruebas de diagnóstico	378	1187	3.1402
Asuntos de seguridad	307	680	2.2149

Métricas

Para la realización de esta tarea se proporcionan algunas métricas para determinar si las palabras claves obtenidas por el algoritmo son palabras claves. Las métricas se definen de la siguiente manera:

Correcta: las coincidencias que corresponden exactamente con las que se encuentran en el archivo *gold (*archivo proporcionado para la correcta evaluación de la tarea*)* tanto en el inicio como en el final de la frase.
Parcial: coincidencias que se reportan cuando entre el intervalo inicio y fin se encuentra una intersección no vacía.
Perdida: coincidencias que aparecen en el archivo *gold,* pero no en *dev* (archivo generado por el algoritmo).
Espuria: coincidencias que aparecen en *dev,* pero no aparecen en *gold*.

Las métricas precisión, exhaustividad, y la medida estándar F_1 están expresadas en términos de las definiciones anteriores [Gutiérrez Vázquez, 2018].

$$Precision = exhaustividad * \frac{correcta + \frac{1}{2}parcial}{correcta + espuria + parcial}$$

$$Exhaustividad = \frac{correcta + \frac{1}{2}parcial}{correcta + perdida + parcial}$$

$$F_1 = 2 * \frac{precision * exhaustividad}{precision + exhaustividad}$$

Resultados

Luego de aplicar el algoritmo a los documentos que conforman el corpus se obtuvieron los resultados que se muestran en las tablas 2 y 3. Por ejemplo, en la tabla 2, el algoritmo obtiene 213 identificaciones correctas, 31 parciales, 137 perdidas y 78 espurias para el documento Asuntos sociales y familiares. El número de identificaciones parciales se debe a que cuando dos palabras claves se solapan, sólo una de ellas se registra como parcial, por lo que la otra se listará como perdida. Por otra parte, las palabras que se identifican como

pérdidas son aquellas que se encuentran en el archivo gold, pero no en dev. Es decir, el algoritmo propuesto no logra generar todas las identificaciones de palabras clave en el texto.

Tabla 2. Número de relaciones correctas, parciales, faltantes o espurias de cada documento.

Documento	*Correct*	*Partial*	*Missing*	*Spurious*
Asuntos sociales y familiares	213	31	137	78
Riñones y sistema urinario	606	110	232	173
Piel, cabello y uñas	803	94	350	219
Asuntos personales de salud	39	13	19	14
Pruebas de diagnóstico	246	59	88	115
Asuntos de seguridad	133	25	82	41

Por otra parte, el algoritmo logra buenos resultados, como puede observarse en la tabla 3, al obtener el 76% de precisión promedio. Asimismo la medida F_1, que puede interpretarse como el promedio ponderado de precisión y exhaustividad, alcanza el valor de 72%, en promedio.

Tabla 3. Resultados de las métricas aplicadas a los documentos que conforman el corpus.

Documento	Precisión	Exhaustividad	F_1
Asuntos sociales y familiares	0.75	0.63	0.68
Riñones y sistema urinario	0.79	0.74	0.77
Piel, cabello y uñas	0.80	0.71	0.75
Asuntos personales de salud	0.76	0.71	0.73
Pruebas de diagnóstico	0.71	0.76	0.73
Asuntos de seguridad	0.78	0.64	0.70
Promedio	0.765	0.6983	0.7267

Sin embargo, si comparamos los resultados obtenidos al aplicar la solución propuesta con los resultados obtenidos mediante la implementación base, véase la tabla 4, podemos observar que alcanza el 78% de precisión, 89% de exhaustividad y 83% de valor F_1 comparados con los valores 76.5%, 69% y 72%, respectivamente.

Tabla 4. Resultados de las métricas obtenidas mediante la implementación base.

Documento	Precisión	Exhaustividad	F_1
Asuntos sociales y familiares	0.76	0.84	0.80
Riñones y sistema urinario	0.76	0.91	0.83
Piel, cabello y uñas	0.77	0.88	0.82
Asuntos personales de salud	0.83	0.90	0.86
Pruebas de diagnóstico	0.76	0.90	0.82
Asuntos de seguridad	0.81	0.93	0.86
Promedio	0.78	0.89	0.83

Como se mencionó anteriormente los resultados experimentales se obtuvieron con los datos de entrenamiento, los datos de prueba estan disponibles en la tarea de 3 del TASS pero no el gold de este segundo conjunto de datos. Por lo que sólo se reportan los resultados obtenidos con los datos de entrenamiento y la implementación base (*baseline*) .

Trabajo a futuro

La propuesta de solución para la tarea planteada también podría resolverse a través del uso de otras metodologías, como el aprendizaje automático para la extracción de las palabras clave. Asimismo, podrían utilizarse y extenderse los modelos estadísticos proporcionados por la biblioteca Spacy para obtener un enfoque que permita mejorar los resultados experimentales, especialmente los valores de exhaustividad y F_1.
Por otra parte, de acuerdo a lo planteado por [Gutiérrez Vázquez, 2018], gracias al desarrollo de esta tarea puede continuarse la realización de las demás tareas planteadas en TASS que incluyen la clasificación de las palabras clave como una acción o concepto y la categorización de las relaciones en relaciones entre conceptos o bien relaciones entre acciones y conceptos o entre acciones.

Conclusiones

Mediante el algoritmo propuesto se generaron archivos que contenían las palabras clave, los cuales se utilizaron para su posterior evaluación mediante una herramienta proporcionada en la tarea. Se obtuvo una precisión de 0.765, exhaustividad de 0.6983 y F_1 de 0.7267, en promedio. La precisión indica que se obtuvo un número menor de identificaciones espurias comparado con el número de identificaciones perdidas. Esto nos muestra que la propuesta de solución obtiene menos relaciones espurias. Sin embargo, el valor de exhaustividad proporcionado, indica que también hubo un valor alto, aunque menor a la precisión, de identificaciones espurias. Así, podría ajustarse el algoritmo para disminuir el número de identificaciones espurias mediante una mejor comprobación de las condiciones contempladas en el mismo puesto que obtiene palabras clave que no se encuentran en el gold, lo que las vuelve espurias. Asimismo, la solución propuesta es competitiva puesto que al compararse con la implementación base sólo hay una diferencia de un 2% en precisión.
Por otra parte, se debe destacar la importancia de trabajos relacionados con el Procesamiento de Lenguaje Natural orientados al dominio de salud y medicina en español puesto que la mayoría de los trabajos están enfocados al idioma inglés. Además, este tema cobrará mayor importancia a futuro pues disponemos de un flujo de información creciente y es indispensable extraer conocimiento significativo de la misma.

Agradecimientos

Esta investigación es apoyada por el Fondo Sectorial de Investigación para la Educación, proyecto CONACYT CB/257357 y por el proyecto VIEP-BUAP.

Referencias

1. Liddy, E.D.; (2001). Natural Language Processing. In Encyclopedia of Library and Information Science, 2nd Ed. NY. Marcel Decker, Inc.
2. Beliga, S.; (2014). Keyword extraction: a review of methods and approaches.
3. Siddiqi, S.; and Sharan, A.; (2015). Keyword and Keyphrase Extraction Techniques: A Literature Revie. International Journal of Computer Applications. **(109)**. 18
4. Stauffer, M.; Fischer, A.; Riesen, K.; (2018). Keyword Spotting in Historical Handwritten Documents based on Graph Matching. *Pattern Recognition*. **(81)**. 240–253.
5. Lynn, H.; Lee, E.; Choi, C.; Kim, P.; (2017). SwiftRank: An Unsupervised Statistical Approach of Keyword and Salient Sentence Extraction for Individual Documents. *Procedia Computer Science*. **(113)**. 472-477.
6. Benny, A.; Mintu, P.; (2015) Keyword Based Tweet Extraction and Detection of Related Topics. *Procedia Computer Science*. **(46)**. 364-371.
7. Gonenc E.; and Ilyas C.; (2007) Using lexical chains for keyword extraction, *Information Processing & Management*. **(43)**. 1705-1714.
8. Honnibal, M.; and Montani, I.; (2017). Spacy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.
9. Bird, S.; Klein, E.; & Loper, E.; (2009). Natural Language Processing with Python. O'Reilly Media Inc. 7-8
10. Gutiérrez Vázquez, Y. (2018). eHealth Knowledge Discovery. TASS. https://tass18-task3.github.io/website/taskA.html