Etiquetado asistido de documentos de investigación mediante procesamiento de lenguaje natural y tecnologías de la web semántica

Assisted labeling of research documents through natural language processing and semantic web technologies

Alfonso Tintinago, Yordan Muñoz, Gustavo A. Uribe, Pedro H. Álvarez Sánchez
Ingeniera de Sistemas, Corporación Universitaria Comfacauca, Popayán, Colombia
Correo-e: guribe@unicomfacauca.edu.co

***Resumen*— El presente artículo se basa en la implementación del procesamiento de lenguaje natural (PLN) y las tecnologías de la web semántica, con la intención de facilitar la extracción de palabras claves en documentos de investigación de forma más eficiente y eficaz. Para tal fin, por medio de una matriz de comparación se seleccionó un algoritmo para realizar el proceso de extracción. Se eligió el algoritmo Keyword Extraction Based On Entropy Difference (C#) realizado por Zhen YANG, Jianjun LEI, Kefeng FAN y Yingxu LAI. Este algoritmo fue desarrollado para procesarlos documentos en idioma chino, por lo que fue requerida una adaptación al idioma inglés y español anexando los vocabularios de correspondientes a estos idiomas configurando el código fuente del algoritmo. Adicionalmente se adaptó el algoritmo para que usase una ontología con la terminología propia del dominio de conocimiento de ingenierías. El algoritmo fue evaluado por medio de ejemplos de artículos científicos, obteniendo métricas de recuperación de la información, como son la precisión, exhaustividad y el valor F. Se obtuvo como resultado un valor F promedio 0.63 para una muestra de 13 artículos científicos, lo que valida el algoritmo como óptimo para la tarea propuesta.**

***Palabras clave*— Metadato, *Recuperación de información*, Procesamiento natural del lenguaje, Web semántica.**

***Abstract*— This article is based on the implementation of natural language processing (NLP) and semantic web technologies, with the intention of facilitating the extraction of keywords in research documents in a more efficient and effective way. For this purpose, an algorithm was selected to perform the extraction process by means of a comparison matrix. The Keyword Extraction Based On Entropy Difference (C #) was chosen by Zhen YANG, Jianjun LEI, Kefeng FAN, Yingxu LAI. This algorithm was developed to execute this process in Chinese language, therefore an adaptation was made to the English and Spanish language by appending the corresponding vocabularies thought the configuration of the source code. Furthermore, the algorithm was adapted for use an ontology that describe the engineering domain. The algorithm was evaluated over of scientific articles, obtaining information retrieval metrics, such as accuracy, completeness and the F value. An average F value 0.63 was obtained for a sample of 13 scientific articles, which validates the algorithm as optimal for the proposed task.**

Key Word — Information retrieval, Metadata, Semantic Web, natural language processing

I. INTRODUCCIÓN

Parte principal para un buen desarrollo de un trabajo científico es la investigación documental de proyectos similares, que se realizan con el fin de obtener conocimientos y guías de estos, como por ejemplo la implementación de metodologías para el desarrollo de los mismos o simplemente referenciar algunas de sus conclusiones. Sin embargo, se evidencia que las herramientas de búsquedas que no cuenta con alta precisión y exhaustividad. Una de las causas probables de este hecho es la escasez de metadatos estandarizados, como las palabras claves. La Web Semántica nos permite hacer uso de vocabularios controlados, por lo que se vislumbra como una solución en conjunto con el procesamiento del lenguaje natural (PLN) para realizar el etiquetado de palabras claves [1].

El campo de estudio que se centra en las interacciones entre el lenguaje humano y los ordenadores se llama procesamiento del lenguaje natural, o PLN. Se encuentra en la intersección de la informática, la inteligencia artificial, y la lingüística computacional. El PLN es una manera para que los ordenadores puedan analizar, comprender y entender el significado del lenguaje humano de una manera inteligente y útil. Mediante la utilización del PLN [2], los desarrolladores pueden organizar y estructurar los conocimientos para realizar tareas tales como el resumen automático, traducción automática, corrección de textos, recuperación de la información, búsqueda de documentos, sistemas inteligentes para la educación y el entrenamiento, reconocimiento de entidades, el análisis de

Fecha de Recepción: 06 de diciembre de 2017
Fecha de Aceptación: 20 de octubre de 2018

opiniones, de reconocimiento de voz, y el tema de la segmentación. [3]

También podemos decir que la *World Wide Web*, basada en documentos y enlaces de hipertexto [4], fue diseñada para la lectura humana y no para que la información que contiene pudiera procesarse de forma automática. Si hacemos una búsqueda de documentos, por ejemplo, por el término "hipertexto", la Web no distingue entre los distintos significados o contextos en los que aparece este término (programas para diseñar hipertexto, información docente, empresas que anuncian su web, etc.). La Web actual tampoco permite automatizar procesos, como, por ejemplo, buscar un seminario sobre hipertexto, hacer la reserva de plaza, consultar los medios de transporte disponibles hasta la ciudad donde se celebre el evento, reservar billete, y conseguir un plano de dicha ciudad. Aun utilizando un potente buscador, se pierden muchas horas navegando por los resultados obtenidos tras la consulta, para acceder a la información de forma manual, cuando esto lo podría hacer un programa o agente inteligente. [5]

La Web Semántica [6], vendría a ser una extensión de la Web actual dotada de significado, esto es, un espacio donde la información tendría un significado bien definido, de manera que pudiera ser interpretada tanto por agentes humanos como por agentes computarizados. Para potenciar el uso de ontologías en la web, se necesitan aplicaciones específicas de búsqueda de ontologías, que indiquen a los usuarios las ontologías existentes y sus características para poder utilizarlas en su sistema. [7]

Con el fin de facilitar el proceso de recuperación de documentos de investigación, se ha planteado la hipótesis ¿Cómo identificar las palabras claves en un documento de investigación teniendo en cuenta una ontología descriptora de un dominio de conocimiento? Como hipótesis se plantea que realizar una solución basada en tecnologías de la web semántica y procesamiento del lenguaje natural (PLN) permitirá la extracción precisa de palabras claves en documentos de investigación. Este etiquetado asistido será evaluado por medio de métricas de recuperación de la información que permitan evaluar la precisión y exhaustividad, con el fin de observar y analizar si este algoritmo es eficiente frente a un etiquetado manual realizado por los autores, para ello se toma como punto de referencia artículos científicos de diferentes áreas como: informática, electrónica, telemática entre otras.

II. CONTENIDO

A. Métodos

Para la identificación del algoritmo se realizó una investigación documental sobre Procesamiento de Lenguaje Natural y las herramientas libres PLN [8], que permitan extraer palabras claves mediante el procesamiento de lenguaje natural y tecnologías de la web semántica en documentos de investigación [9]. Además, se investigaron referentes teóricos y metodologías a implementar, que permitieron tener una visión más clara sobre dichos temas. Para la selección del algoritmo se realizó una matriz comparativa sobre distintos tipos de propuestas que permitan dicha extracción (siguiente sección). La ontología utilizada para la adaptación del algoritmo fue desarrollada en base a palabras claves del conocimiento de la ingeniería teniendo en cuenta temas de investigación relevantes como los presentes en Scimago y la taxonomía estandarizada de la UNESCO.

B. Proceso de selección del algoritmo de extracción de palabra claves

Con el fin de seleccionar el algoritmo más apropiado se tuvieron en cuenta aspectos como: su disponibilidad, precisión y disponibilidad de herramientas. Dicho análisis dio como resultado la Tabla 1, en donde finalmente fue seleccionado el algoritmo Keyword Extraction Based On Entropy Difference (C#) realizado por Zhen YANG, Jianjun LEI, Kefeng FAN, Yingxu LAI [10].

Tabla 1: Criterios de evaluación para la selección del algoritmo.

Comparación de algoritmos			
Nombre del algoritmo	**Descripción**	**Aspectos Positivos**	**Aspectos Negativos**
Rapid automatic keyword extraction (Rake) (Python). [11]	La Extracción de Palabra clave Rápida Automática (el RAKE) el algoritmo extrae palabras clave del texto, por medio de que va identificando en el texto las palabras que no son vacías y luego hace la notación de estas frases a través del documento. Esto no requiere ninguna educación, la única entrada es una lista de palabras de parada para una lengua dada, y un tokenizador que divide el texto en frases y esas frases en palabras.	RAKE es una biblioteca simple extracción de palabras clave que se centra en la búsqueda de frases de varias palabras que contengan palabras frecuentes. Sus puntos fuertes son su simplicidad y la facilidad de uso, ya que se implementa la librería en el entorno de desarrollo de Python haciendo más fácil su uso.	En cuanto a la desventaja a la utilización del algoritmo RAKE, se tiene como puntos débiles: precisión limitada, el requisito de configuración de parámetros, y el hecho de que descarta muchas frases válidas y no normaliza candidatos.
Extracción de palabras	*Maui significa multiusos tema automático de indexación. Es una biblioteca con licencia GPL.* Maui identifica automáticamente los temas principales en los documentos de texto. Dependiendo de la tarea, los temas son	En comparación con RAKE, Maui le permite al usuario: • Extracción de las palabras clave no sólo de texto, sino	En el momento de la utilización de Maui en java se obtienen errores en cuanto a la implementación de los modelos de formación para

clave utilizando algoritmo Maui (Java). [12]	etiquetas, palabras clave, frases, palabras de vocabulario, descriptores, términos de índice o títulos de los artículos de Wikipedia.	también con una referencia a un vocabulario controlado. • Mejorar la precisión mediante la formación de Maui en las palabras clave elegidas manualmente	el vocabulario controlado, perdiéndose la construcción de los modelos para la extracción de las librerías de Wikipedia MINER, por lo cual no se extrae correctamente las palabras claves (metadatos)
Keyphrase Extraction Algorithm (KEA) (Java). [13]	Es un algoritmo para la extracción de palabras clave de documentos de texto. Se puede utilizar ya sea para la indexación libre o para la indexación con un vocabulario controlado. KEA está implementado en Java y es independiente de la plataforma. Es un software de código abierto distribuido bajo la Licencia Pública General de GNU.	El algoritmo de extracción de frases claves tiene como ventajas: fácil de instalar y utilizar, directamente desde código o desde la línea de comandos, con cualquier vocabulario de texto o formato SKOS, últimas bibliotecas, incluyendo Jena-2.4 y Weka-3.5.5, fácilmente aplicable a los nuevos lenguajes y dominios distribuido con vocabularios e la muestra en 3 idiomas (inglés, español, francés), contiene documentos de la muestra en 3 idiomas para crear y probar los modelos	Se utiliza el algoritmo para realizar la extracción de los metadatos o frases claves, pero con una indexación libre que limita la extracción precisa y exhaustiva.
Term frequency – Inverse document frequency. **(TF-IDF algorithm)**	Hace uso de herramientas estadísticas que pretenden reflejar la importancia de una palabra en un documento o corpus. Las variaciones del esquema de ponderación TF-IDF a menudo son utilizadas por los motores de búsqueda como una herramienta central en la puntuación y la clasificación de un documento de relevancia. TF-IDF se puede utilizar con éxito para palabras de filtrado en varios campos, incluyendo el resumen de texto y clasificación	El uso del algoritmo TF-IDF es importante ya que usa la frecuencia de los términos, en frecuencia inversa de documento. Es una manera de marcar la importancia de las palabras (o "términos") en un documento basado en la frecuencia con que aparecen en varios documentos. Si una palabra aparece con frecuencia en un documento, que es importante, se le da una alta puntuación. Pero si una palabra aparece en muchos documentos, no implica relevancia. Por lo tanto, las palabras comunes como "el" y "para", que aparece en muchos documentos, serán mal ponderadas.	Al momento de la implementación del algoritmo TF-IDF, el documento a analizar se debe introducir dentro del algoritmo o sea que se debe copiar todo el texto del documento dentro de las variables del código, generando así mucha información dentro del algoritmo, desmejorando la calidad de la precisión al momento de extraer los metadatos, ya que este algoritmo no cuenta con la forma de leer los archivos cargándolos en archivos con extensión txt.
Keyword Extraction Based On Entropy Difference (C#). **(Algoritmo seleccionado)**	En este programa, se utiliza una especie de entropía media para extraer las palabras clave en un texto. Es una medida sencilla, sin ninguna información a priori y extraer con eficacia las palabras clave en un solo texto. Mediante el uso de este software, la única cosa que hay que hacer es establecer la ruta de los archivos que se requiere, y el software puede ayudar a terminar el resto del trabajo. A continuación, puede seleccionar uno de los métodos de entropía General o de máxima entropía para extraer las palabras clave.	Como un nuevo algoritmo en el campo de la extracción de palabras clave, este algoritmo tiene los siguientes puntos destacados: • Es una nueva métrica para evaluar y clasificar la relevancia de las palabras en un texto. • La métrica utiliza la diferencia de entropía de Shannon entre el modo intrínseco y extrínseco. • Este trabajo es un nuevo resultado en la extracción de palabras clave y el ranking. • Este método es especialmente adecuado para documentos individuales de las cuales no hay una información a priori disponible.	La comprensión de la complejidad del texto escrito humana requiere un análisis adecuado de la distribución estadística de las palabras en los textos. Se encuentra con palabras muy significativas tienden a ser modulada por la intensión del autor de la escritura, mientras que las palabras comunes se extienden esencialmente de manera uniforme en un texto.

C. Adaptación del algoritmo

Seleccionado el algoritmo se procedió a su implementación, para ello se debió realizar la respectiva adaptación al idioma inglés y español, dado que por defecto el sistema es funcional en el idioma chino. Primeramente, se anexan los dos listados de vocabularios con los idiomas respectivos en formatos dll, para que sean adaptados y leídos por el algoritmo mediante la indexación de una biblioteca de clases, y así sean tenidos en cuenta por la clase WORDFRE al momento de realizar la diferencia de entropía. Adicionalmente, se crean dos listas con archivos planos en donde se incorporan las *stopwords* para que el algoritmo las tenga en cuenta (Figura 1), ya que no generan ningún tipo de información relevante al realizar el análisis del texto, procediendo a eliminarlas, con el fin de hacer más fácil la extracción de palabras claves en los documentos de investigación.

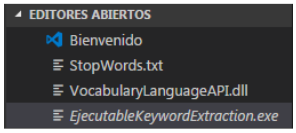

Figura 1: Biblioteca de clases de cambio de idioma dll y lista de *stopwords.*

Los documentos de investigación son convertidos en archivos planos con extensión txt, para que sean leídos por la aplicación final del algoritmo, a continuación, se realiza una normalización del texto por parte de la herramienta, donde esta función se utiliza para estandarizar el documento original. El proceso de normalización incluye la eliminación de saltos de línea, espacios múltiples consecutivos y signos de puntuación. La función *removestop,* utiliza los archivos planos que contienen las stopwords para eliminar las palabras vacías del documento seleccionado. Seguidamente después de cumplir con los procesos anteriores se procede a la utilización de la clase WORDSFRE, que se encarga de dividir el documento en palabras, con el fin de encontrar con qué frecuencia está una palabra en el documento, la ubicación, la distancia entre dos palabras sucesivas y la diferencia de entropía entre ellas.

Se realiza la respectiva extracción de las palabras claves en documentos de investigación con extensión .txt, utilizando la teoría de la entropía de Shannon, la cual mide la incertidumbre de una fuente de información [14], adicionalmente el sistema es enriquecido con la implementación de la ontología, la cual va a contener un listado de palabras clave con respecto a los temas más relevantes en los documentos en el dominio de las ingenierías. Para agregar la ontología al código fuente del algoritmo se procede a la adaptación de un archivo en formato XML el cual fue creado por una estudiante de decimo semestre de ingeniería de sistemas de la Corporación Unicomfacauca, éste contiene el nombre de los autores, títulos y tipo de documentos de investigación más relevantes en la investigación de ingeniería. Este XML se adapta por medio de la creación de un archivo con extensión dll para ser indexado por medio de una referencia de biblioteca de clases, permitiendo ser utilizado por el algoritmo en el entorno de desarrollo C#, el cual es lenguaje en el que se realizó el sistema [15] (ver Figura 2).

```
2   using System.Collections.Generic;
3   using System.Linq;
4   using System.Text;
5   using System.Threading.Tasks;
6   using System.Xml;
7
8   namespace ConsoleApplication1
9   {
10      class Program
11      {
12          static void Main(string[] args)
13          {
14
15              try
16              {
17                  XmlDocument xmldoc = new XmlDocument();
18                  string file1 = @"c:\final 11-10-16.xml";
19                  xmldoc.Load(file1);
20                  xmldoc.Save(Console.Out);
21                  Console.Read();
22              }
23              catch (Exception e)
24              {
```

Figura 2: Codificación para leer el archivo con la ontología (XML)

En la Figura 3 se presenta un ejemplo de extracción de las palabras claves en un documento de investigación sin haberse implementado la ontología y posteriormente se realiza el mismo procedimiento al documento, pero ya implementada la ontología en el algoritmo (Figura 4);

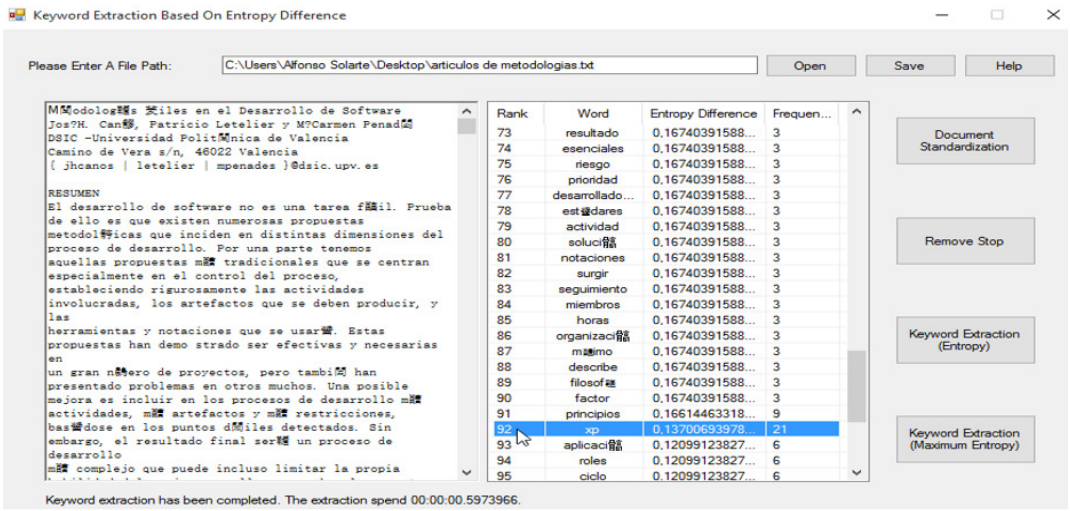

Figura 3: Extracción de palabras claves del documento sin la implementación de la ontología.

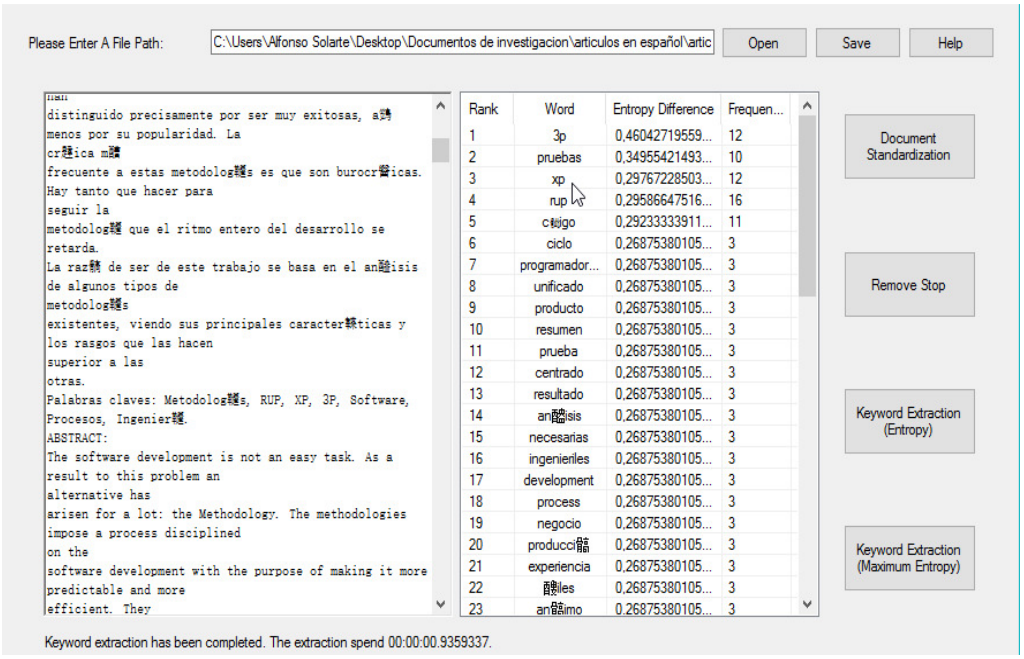

Figura 4: Extracción de palabras claves del documento con la implementación de la ontología.

Se observa en la figura 4, que, al implementar la ontología, se le da mayor importancia a las palabras que ésta contiene, como, por ejemplo: Pruebas, XP y RUP; que son palabras propias del dominio de la ingeniería, ponderándolas en los primeros lugares de la lista al momento de realizar la extracción de las palabras claves en el algoritmo.

La teoría de la entropía de Shannon, considera la cantidad de información promedio que contienen los símbolos usados. Los símbolos con menor probabilidad aportan mayor información; por ejemplo, si se considera como sistema de símbolos a las palabras en un texto, palabras frecuentes como «que», «el», «a» aportan poca información, mientras que palabras menos frecuentes como «corren», «niño», «perro» aportan más información. Si de un texto dado borramos un «que», seguramente no afectará a la comprensión y se sobreentenderá, no siendo así si borramos la palabra «niño» del mismo texto original. Cuando todos los símbolos son igualmente probables (distribución de probabilidad plana), todos aportan información relevante y la entropía es máxima. [16].

D. Resultados

Para evaluar los resultados, se usan las métricas de recuperación de la información tales como la precisión, exhaustividad y el valor F, con el objetivo de evaluar y analizar las palabras claves arrojadas

por el sistema comparándolas con las que el autor del documento refiere.
En la ecuación (1) encontramos la fórmula matemática para la precisión y en la ecuación 2 para la exhaustividad.

Según Salton [17], la precisión es el porcentaje de datos relevantes recuperados entre el número total de datos recuperados. Para nuestra propuesta la formula equivalente se muestra a continuación.

$$Precisión = \frac{\#de\ palabras\ claves\ correctas}{\#\ total\ de\ palabras\ claves\ obtenidas} x100 \quad (1)$$

Según Swets, J. A [18], expresar la proporción de datos relevantes recuperados, comparado con el total de los datos que son relevantes existentes en la base de datos, con total independencia de que éstos, se recuperen o no. Para nuestra propuesta la formula equivalente se muestra a continuación.

$$Exhaustividad = \frac{\#\ de\ palabras\ claves\ correctemente\ recuperadas}{\#\ palabras\ claves\ definidas\ por\ el\ autor} x100 (2)$$

Cabe denotar que cuanto más se acerque el valor de la precisión al valor nulo (0), será mayor el número de palabras claves recuperados que no se consideren relevantes. Por el contrario, si el valor de la precisión es igual a uno (1), se juzgará que todas las palabras claves recuperadas son relevantes.

La exhaustividad evalúa que tan completo fue el etiquetado, por ejemplo, sobre un documento que contienen 10 palabras claves y como resultado de esta búsqueda solo se obtienen 6, es decir que si aplicamos la formula anterior se puede observar que el índice de exhaustividad es del 60%.

En efecto el resultado de la formula anterior arroja valor uno (1), se obtendrá la exhaustividad máxima posible, lo cual indica que se ha encontrado todas las palabras claves que son relevantes en el documento de investigación. Pero si en algún caso el que el valor de la exhaustividad sea igual a cero (0), se tiene que las palabras obtenidas no poseen relevancia alguna. [19]

En teste proyecto se tiene como prioridad la precisión más que la exhaustividad debido a que el número de palabras claves frecuentemente es reducido y por ende no es muy relevante la inclusión de todas las palabras. Por lo tanto en le cálculo del valor f (ecuación 3) * , se le dará mayor prioridad a la precisión con un valor β = 1/4.

$$F_\beta = (1+\beta^2).\frac{Precisión.Exhaustividad}{(\beta^2.Precisión)+Exhaustividad} (3)$$

Si β es igual a uno, se está dando la misma ponderación (o importancia) a Precisión que, a la Exhaustividad, si β es mayor que uno de damos más importancia a Exhaustividad, mientras que si es menor que uno se le da más importancia a la Precisión. [20]

El procedimiento de la evaluación de las pruebas se realizará tomando como referencias documentos, artículos y revistas en el campo tecnológico, informático y electrónico de autores al azar, con el fin de poder evaluar la extracción de las palabras claves en un etiquetado asistido, echo por la herramienta con el algoritmo frente al etiquetado manual realizado por los autores de los documentos. Como se ha dicho que los documentos que se trabajaron para realizar la evaluación, estarán en el formato que trabaja la herramienta los cuales son archivos planos txt. Además, se debe tener en cuenta que la extracción de las palabras claves debe tener una frecuencia mayor o igual a 10 para realizar el respectivo análisis.

A continuación, se presenta ejemplos donde se desarrolla la evaluación de los documentos o artículos descargados de la web, procesados en la herramienta con el algoritmo tomando como referencia los realizados por los autores manualmente.

Ejemplo a) Título del libro: Ingeniería de Sistemas Realidad Virtual (Figura 7).

Palabras clave: Educación, aprendizaje, pensamiento sistémico, Ingeniería de Sistemas, realidad virtual, modelamiento, dinámica de sistemas y simulación.

Palabras totales del documento: 4053
Palabras en total encontradas por el algoritmo: 13

Etiquetado asistido

Rank	Word	Entropy Difference	Frequen...
1	simulaci[illegible]	0,63381054678...	21
2	virtual	0,47570779232...	18
3	real	0,44223074431...	15
4	natural	0,41107630982...	10
5	elementos	0,38326265690...	17
6	sistema	0,38106281846...	25
7	aprendizaje	0,37808438238...	50
8	realidad	0,36194466176...	40
9	comportami...	0,35355035218...	24
10	mentales	0,33295033281...	29
11	idea	0,31741378786...	12
12	sistemas	0,29010798967...	39
13	mundo	0,28184150683...	14
14	proceso	0,27833720889...	24
15	decisiones	0,27707875482...	9
16	ciclo	0,26539124885...	7
17	matem[illegible]ico	0,25845926469...	13
18	modelamiento	0,24212091880...	19
19	artificial	0,22879157404...	9
20	art[illegible]ulo	0,22605579089...	8
21	diferentes	0,22605579089...	8
22	formal	0,22605579089...	8
23	variable	0,22605579089...	8

Figura 7: Palabras claves obtenidas por el sistema (ejemplo a).

Para este ejemplo se obtuvieron los siguientes valores de precisión y exhaustividad.

$$Precisión = \frac{9}{13} = 0.69$$

$$Exhaustividad = \frac{9}{11} = 0.81$$

Tomando el ejemplo anterior, se tiene que con el algoritmo obtuvo una exhaustividad del 0.81 o de un 81%, por consiguiente, la extracción de las palabras claves es bastante exhaustiva cuando se realiza con la herramienta, sin embargo, la precisión es un poco menor pero dentro del rango aceptable. Con base a estos valores se obtiene el Valor-F que se muestra a continuación:

$$F_\beta = \left(1 + \left(\frac{1}{4}\right)^2\right) x \frac{0.69x0.81}{\left(\left(\frac{1}{4}\right)^2 x0.69\right) + 0.81}$$

$$F_\beta = 0.68$$

El valor obtenido del Valor-F es aceptable para este caso.

Ejemplo b) Título del libro: Dialnet-Los Artículos Científicos Como Metodología De Aprendizaje (Figura 8).

Palabras clave: Artículos científicos, metodología, aprendizaje, ingeniería de sistemas.

Palabras totales del documento: 3227
Palabras en total encontradas por el algoritmo: 8

$$Precisión = \frac{6}{8} = 0.75$$

$$Exhaustividad = \frac{6}{6} = 1$$

Con los resultados anteriores se puede observar que la exhaustividad del etiquetado asistido es la máxima para este caso, incluyendo todos los resultados correctos posibles.

Etiquetado asistido

Rank	Word	Entropy Difference	Frequen...
1	semestre	0,43836019905...	21
2	estudiantes	0,41695863596...	24
3	datos	0,37378033456...	16
4	estrategia	0,34454509796...	10
5	aprendizaje	0,34454509796...	10
6	cient[illegible]tico	0,29300930432...	9
7	investigaci[illegible]	0,28374455201...	16
8	contacto	0,28374455201...	16
9	art[illegible]ulos	0,26525542007...	36
10	contenido	0,24471449654...	10
11	hallazgos	0,23772903962...	8
12	publicaci[illegible]	0,23772903962...	8
13	proceso	0,23123866588...	10
14	redacci[illegible]	0,23123866588...	10
15	publicar	0,22423584070...	3
16	mundo	0,22423584070...	3
17	obtenido	0,22423584070...	3
18	consulta	0,22423584070...	3
19	aproximaci[illegible]	0,22423584070...	3
20	docentes	0,22423584070...	3
21	software	0,22423584070...	3
22	bruner	0,22423584070...	3
23	fundamento	0,22423584070...	3

Figura 8: Análisis de etiquetado asistido (ejemplo b).

A continuación, se muestra el Valor f aplicado a los documentos analizados, enfatizando en la precisión:

$$F_\beta = \left(1 + \left(\frac{1}{4}\right)^2\right) x \frac{0.75x1}{\left(\left(\frac{1}{4}\right)^2 x0.75\right) + 1}$$

$$F_\beta = 0.76$$

Este Valor F es cercano al 80% y corresponde a un caso exitoso, comparándolo con la precisión, exhaustividad y Valor-F, reportados por los autores del algoritmo en el idioma chino. Estos valores de referencia son:

$$Precisión = \frac{7}{10} = 0.70$$

$$Exhaustividad = \frac{7}{9} = 0{,}77$$

$$F_\beta = \left(1 + \left(\frac{1}{4}\right)^2\right) x \frac{0.70x0.77}{\left(\left(\frac{1}{4}\right)^2 x0.70\right) + 0.77}$$

$$F_\beta = 0.70$$

En total se realizaron pruebas a 13 documentos y artículos de investigación en el campo tecnológico, informático, electrónico y telemático, en el idioma de español e inglés de autores al azar. La precisión osciló entre 0.5 y 1, teniendo un valor promedio de 0.61, y con respecto a la exhaustividad los resultados fueron aún mejores ya que se tuvo un promedio de 0.8

(Figura 9). El valor F, calculado con un β = ¼, tuvo de igual manera buenos resultados teniendo un valor promedio de 0.63.

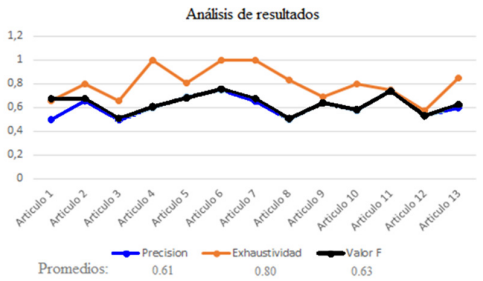

Figura 9: Promedios de precisión, exhaustividad y valor F.
Fuente: Elaboración Propia

Dado los resultados positivos obtenidos en la precisión, exhaustividad y valor F, se deduce que traerá beneficios óptimos como facilitar el proceso de extracción de palabras clave en documentos de investigación, haciendo más eficaz este proceso.

El algoritmo podría ser consumido por diferentes aplicaciones o plataformas relacionadas con el procesamiento del lenguaje natural y tecnologías de la web semántica, con el fin de realizar de una forma más fácil la extracción de palabras claves en diferentes documentos.

III. CONCLUSIONES

Con la implementación de algoritmos basados en el procesamiento de lenguaje natural y tecnologías de la web semántica, permitió realizar la extracción de metadatos como las palabras claves encontradas en documentos de investigación por medio de un etiquetado asistido, el cual fue analizado y evaluado por métricas de recuperación de la información centradas específicamente en la precisión y exhaustividad; este etiquetado fue comparado con el etiquetado manual realizado por los autores en sus documentos obteniendo un valor F promedio de 0.63 y una exhaustividad de 0.8.

El PLN, fue desarrollado y establecido a partir de la experiencia de las personas en cuanto a la facilidad de comunicación con la computadora, puede ser utilizado para analizar contextos como: resumen y traducciones automáticas, corrección de textos, recuperación de la información, búsqueda de documentos, reconocimiento de voz, análisis de opiniones y para sistemas inteligente basados en educación y entrenamiento. A demás el PLN consiste en la implementación de un lenguaje natural para comunicarnos con la computadora de manera más entendible y natural, ya que permite entender las oraciones que le sean suministradas. El uso del PLN facilita la ejecución de programas que cumplan tareas relacionadas con el lenguaje.

Entender la complejidad del texto escrito por el ser humano requiere un análisis apropiado de la distribución estadística de las palabras en los textos. Encontramos que las palabras altamente significativas tienden a ser moduladas por la intensión del escritor, mientras que las palabras comunes se distribuyen uniformemente en el texto. Las ideas de este trabajo se pueden aplicar a cualquier idioma, mediante un vocabulario controlado, sin requerir ningún conocimiento previo sobre semántica o sintaxis.

Por último, es de resaltar que la inclusión del vocabulario controlado en forma de ontología con palabras específicas de un dominio de conocimiento, para este caso de ingenierías, mejoró la identificación de palabras clave. Debido a que las palabras de mayor relevancia que son las que contiene la ontología, el sistema las posiciona en la parte superior de la lista.

REFERENCIAS

[1] Pedro Reyes Columé., "Problemas de Etiquetado Complejidad Computacional", Dpto de matemática aplicada-Universidad de Sevilla, Sevilla 2002.

[2] Matt Kiser. "Product manager at Algorithmia http://blog.algorithmia.com/introduction-natural-language-processing-nlp/, (2016)

[3] Acebrón García, Sergio (2012). "Procesamiento del lenguaje natural en la recuperación de la información". Universidad Carlos III de Madrid. Departamento de Ingeniería Telemática.

[4] Adelaide Bianchini – Conceptos y definiciones de hipertexto. Dpto. de Computación y Tecnología de la Información – Universidad Simón Bolívar, Caracas 1999.

[5] Tim Berners-Lee, James Hendler, *"Ora Lassila, the Semantic Web", Scientific American, May 2001.*

[6] Berners-Lee, T.; Hendler, J.; Lassila, O. "The semantic web: a new form of web content that is meaningful to computers will unleash a revolution of new possibilities". En: Scientific American, 2001.

[7] Arpirez J., Gómez-Pérez A., Lozano Tello A. and Pinto S. "Reference Ontology and (Onto) 2Agent: The Ontology Yellow Pages", Knowledge and Information Systems, an International Journal, SpringerVerlag, 2 (2000) 4, 387-412. Mar. 2000.

[8] Alonso i Alemany, L. Herramientas Libres para Procesamiento del Lenguaje Natural. 5tas Jornadas Regionales de Software Libre, Rosario, Argentina, (2005).

[9] Gerado Sierra. Búsqueda de palabras a partir de las definiciones en los diccionarios de lengua automatizados. Simposios Internacionales de Comunicación Social, Simposio 7, Actas 2, Santiago de Cuba, 2001.

[10] Zhen YANG, Jianjun LEI, Kefeng FAN, Yingxu LAI. Keyword Extraction by Entropy Difference between the Intrinsic and Extrinsic Mode, Physica A: Statistical Mechanics and its Applications, 392 (2013), 4523-4531

[11] Rose, S., Engel, D., Cramer, N., & Cowley, W. (2010). Automatic Keyword Extraction from Individual Documents. In M. W. Berry & J. Kogan (Eds.), Text Mining: Theory and Applications: John Wiley & Sons.

[12] Jones, S. and Paynter, G.W. (2001). "Human evaluation of Kea, an automatic keyphrasing system". *First ACM/IEEE-CS Joint Conference on Digital Libraries*, Roanoke, Virginia, June 24-29, 2001, ACM Press, pp148-156.

[13] Alyona Medelyan as a part of her PhD project under supervision of Ian H. Witten and Eibe Frank. (2014). Extracción de palabras clave utilizando algoritmo Maui. Department of Computer Science at the University of Waikato, New Zealand.

[14] Cuevas Agustín, Gonzalo, "Teoría de la información, codificación y lenguajes", Ed. SEPA (Sociedad para Estudios Pedagógicos Argentinos), Serie Informática 1986.

[15] Corcho O., Gómez-Pérez A. "A Roadmap to Ontology Specification Languages". EKAW 2000 pp. 80-96.

[16] Jorge Ramió Aguirre, Aplicaciones criptográficas. Libro guía de la asignatura de Seguridad Informática. Escuela Universitaria de Informática. Universidad Politécnica de Madrid. Enero 1998.

[17] Salton, G. y M. J. McGill., (1983), Introduction to Modern Information Retrieval. New York: McGraw Hill.

[18] Swets, J. A., (1963), Information retrieval Systems. Science, 141 (3577): July 1963 p. 245-250.

[19] Kent A. Et al., (1955), Machine literature searching. VIII. Operational Criteria for Designing Information Retrieval Systems American Documentation Abril, 6 (2) p. 93-101.

[20] Beitzel., Steven M. (2006). On Understanding and Classifying Web Queries (Ph.D. thesis). IIT. CiteSeerX: 10.1.1.127.634.

[21] Hripcsak, G., & Rothschild, A. S. (2005). Agreement, the f-measure, and reliability in information retrieval. Journal of the American Medical Informatics Association, 12(3), 296-298.