INSTITUTO POLITÉCNICO NACIONAL
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

Laboratorio de Ciencia de los Datos y Tecnología de Software

Tesis:

"Modelo de detección de cúmulos naturales basado en una taxonomía semántica"

Que para obtener el grado de:
"Maestría en Ciencias de la Computación"

Presenta:

Ing. Víctor Uriel Zaragoza Luna

Directores de tesis:

Dr. Gilberto Lorenzo Martínez Luna
Dr. Jesús Manuel Olivares Ceja

Ciudad de México, junio 2016

SIP-14 bis

INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México siendo las 10:00 horas del día 30 del mes de mayo de 2016 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis titulada:

"Modelo de detección de cúmulos naturales basado en una taxonomía semántica"

Presentada por el alumno:

ZARAGOZA	**LUNA**	**VÍCTOR URIEL**
Apellido paterno	Apellido materno	Nombre(s)

Con registro:

B	1	4	0	5	7	5

aspirante de: **MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN**

Después de intercambiar opiniones los miembros de la Comisión manifestaron ***APROBAR LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA
Directores de Tesis

Dr. Gilberto Lorenzo Martínez Luna

Dr. Jesús Manuel Olivares Ceja

Dr. Adolfo Guzmán Arenas

Dr. René Luna García

Dr. Grigori Sidorov

PRESIDENTE DEL COLEGIO DE PROFESORES

INSTITUTO POLITECNICO NACIONAL
CENTRO DE INVESTIGACION
EN COMPUTACION
DIRECCION

Dr. Luis Alfonso Villa Vargas

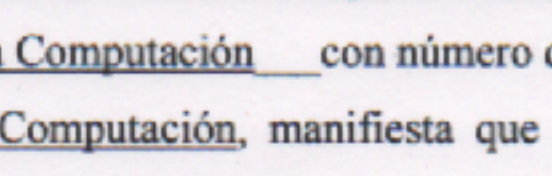
INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de _____México_____el día __7__del mes_____junio_____del año 2016, el que suscribe____Víctor Uriel Zaragoza Luna_____ alumno del Programa de_Maestría en Ciencias de la Computación____con número de registro _B140575_, adscrito al Centro de Investigación en Computación, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección de Dr. Gilberto Lorenzo Martínez Luna_ y _el Dr. Jesús Manuel Olivares Ceja y cede los derechos del trabajo intitulado Modelo de Detección de Cúmulos Naturales basado en una Taxonomía Semántica, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección victor.uriel.zaragoza.luna@gmail.com. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.



Víctor Uriel Zaragoza Luna

Nombre y firma

Resumen

El trabajo presente describe un sistema que estima el número de cúmulos naturales (aquéllos que normalmente una persona hallaría). Los cúmulos se obtienen con base en la mayor relación semántica de los términos de las noticias en idioma español que se usan como caso de estudio. En cada cúmulo obtenido se asignan etiquetas representativas utilizando los términos con mayor peso en cada cúmulo. También, se desarrolló una forma de visualización basada en los parámetros utilizados para detectar los cúmulos, en este caso, la inter distancia, la intra distancia, la cardinalidad de cada cúmulo y sus términos más representativos.

El sistema utiliza documentos (información no estructurada) de noticias recolectadas de sitios web públicos (prensa digital en español). El contenido de cada noticia se homogeneiza mediante las técnicas de lenguaje natural: remover las palabras con valor semántico mínimo (stop-words), lematización y aplicación de sinónimos. El resultado de esta etapa se procesa con el algoritmo LDA (Latent Dirichlet Allocation) para encontrar los tópicos principales que están implícitos en los documentos y al mismo tiempo localizar los términos que se consideran equivalentes en forma semántica latente. En la etapa siguiente, los términos obtenidos se traducen al idioma inglés para buscar sus equivalencias semánticas utilizando WordNet. En la siguiente etapa se utiliza la distancia Path [39] para calcular las distancias que existen entre las palabras de un cúmulo (intra-distancias) y las distancias entre cúmulos (inter-distancias) para encontrar la compacidad y la lejanía que se utilizan para identificar el número de cúmulos naturales del conjunto de datos.

La selección del número más "natural" (k) de agrupaciones de los documentos que se procesan, se hace con base en dos criterios: maximizar la inter-distancias entre cúmulos y minimizar las intra-distancias entre agrupaciones.

Se hicieron pruebas con documentos de noticias de diferentes temas sociales para validar la correcta agrupación y determinación del número k obtenido, entre estos están: reciclaje de desperdicios, deportes, política y salud. Los resultados obtenidos permiten considerar esta tesis como una alternativa para estima el número de cúmulos de una colección de documentos.

Abstract

This work describes a system that estimates the number of natural clusters (that usually a person could find). The clusters are obtained based on the greatest semantic relationship among the terms of documents used as study case. Each document contains news in Spanish. Each obtained cluster is labeled using the most representative terms in each cluster (the highest weight). It was developed a visualization schema that reflects the parameters used to detect clusters, in this case, the inter distance, intra distance, the cardinality of each cluster and the most representative terms.

The system uses documents (unstructured data) collected from news public websites (Spanish digital press). The content of each news is homogenized using natural language techniques: removing stop-words, stemming and finding synonyms. The result of this step, is then processed with the LDA (Latent Dirichlet Allocation) algorithm to find the main topics that are embedded in documents and at the same time find out the terms that are considered latent semantically equivalent. The next step takes the words obtained and translate them into English to find their semantic equivalence using WordNet. Then, Path [39] is used to calculate the distances between cluster words (intra-distances) and the distances between clusters (inter-distances) to obtain the compactness and the distance that are then used to identify the number of natural clusters of the used data set.

The selection of the most "natural" number (k) of document groups is based on two criteria: maximize inter-distances between clusters and minimize intra-distances in clusters.

Several tests were conducted with news from different social topics to validate if the number k obtained corresponds to the content of the collection. The topics were: waste recycling, sports, politics and health.

The results allow us to consider this thesis as an alternative to estimate the number of clusters of a document collection.

Agradecimientos

En primera instancia agradezco profundamente a toda mi familia, especialmente a mis hermanos y a las únicas dos personas que admiro en este mundo mis amados padres “Norma y Víctor”, ya que sin su apoyo simplemente mis logros no existirían.

Al Dr. Gilberto Luna, por apoyarme en una de las etapas más obscuras de mi vida y darme la oportunidad de mostrar mi valía, y sobre todo por darme la confianza para involucrarme en todos sus proyectos y con ello fortalecer mi formación.

Al Dr. Adolfo Guzmán Arenas, cuyos consejos, asesoramiento y apoyo me acompañaron en todo este trabajo, pero primordialmente por inspirarme a no detenerme, a nunca parar mientras se tenga una idea para materializar y a que los límites están solamente en el lugar en que unos los coloque.

Al Dr. Jesús Olivares, por siempre tener la frase clave de apoyo para continuar.

A los pocos, pero muy valiosos amigos que me encontré a lo largo del camino y que de una u otra forma se convirtieron en mis maestros.

Índice

Glosario

Cadenas léxicas [1]
Se desprenden del área de cohesión semántica en lingüística. La cohesión se encarga de la relación entre palabras que conectan diferentes fragmentos de un texto. Una cadena léxica es una secuencia de palabras relacionadas que dan pistas importantes sobre la semántica del contenido del texto, así, permite identificar los temas principales en documentos.
Cúmulo («cluster») [42]
Es un arreglo de artículos cuyas características son similares entre sí y distintos a otros cúmulos.
Conceptos denotados por más de una palabra [2]
Concepto formado por dos o más palabras, por ejemplo: Nuevo México (dos palabras). Denotan el concepto "Nuevo-México", un estado de EE. UU; Benito Juárez (dos palabras). Denotan al Benemérito de las Américas; los Tres Mosqueteros (tres palabras). Denota un libro de Alejandro Dumas.
Jerarquía [3]
Para un elemento conjunto E, una jerarquía H de E, es un elemento conjunto en el cuál cada elemento e_i es un valor simbólico que representa dos cosas, un valor único de E o una partición y $\cup_i \{r_i \mid e_i \propto r_i\} = E$ (La unión de todos los conjuntos representados por e_i forman E).
Ejemplo, jerarquía H_1: para E = {*Canada, USA, Mexico, Cuba, Puerto_Rico, Jamaica, Guatemala, Honduras, Costa_Rica*}={a, b, c, d, e, f, g, h, i}, una jerarquía H1 es {*North_America, Caribbean_Island, Central_America*}={H_1^1, H_1^2, H_1^3}, donde *North_America* ∝{*Canada, USA, Mexico*}; *Caribbean_Island* ∝ {*English_Speaking_Island, Spanish_Speaking_Island*}={H_1^{21}, H_1^{22}}; *English_Speaking_Island* ∝ {*Jamaica*}; *Spanish_Speaking_Island* ∝ {*Cuba, Puerto_Rico*}; *Central_America* ∝ {*Guatemala, Honduras, Costa_Rica*}.
Lexema [43]
Unidad léxica abstracta que no puede descomponerse en otras menores, aunque sí combinarse con otras para formar compuestos, y que posee un significado definible por el diccionario, no por la gramática.
Partición
La división de un conjunto en un número finito de subconjuntos (llamados cúmulos o clusters en inglés) que son mutuamente excluyentes y colectivamente exhaustivos.
Patrón
En el área de la computación definida como Reconocimiento de Patrones se considera un patrón a la representación simbólica o abstracta de los atributos percibidos en cualquier ente, objeto, fenómeno, suceso o secuencia de ellos pertenecientes al mundo real o abstracto. Dicha representación se obtiene generalmente de modelar sus características en forma de tupla o vector de rasgos [45].

RSS (Really Simple Syndication)
Es un formato XML para sindicar o compartir contenido en la web. Se utiliza para difundir información actualizada frecuentemente a usuarios que se han suscrito a la fuente de contenidos [61].

Similitud
Parecido entre dos datos simbólicos: (que tan lejos está la palabra “perro” de “caballo” o “dinero” de “euro”).

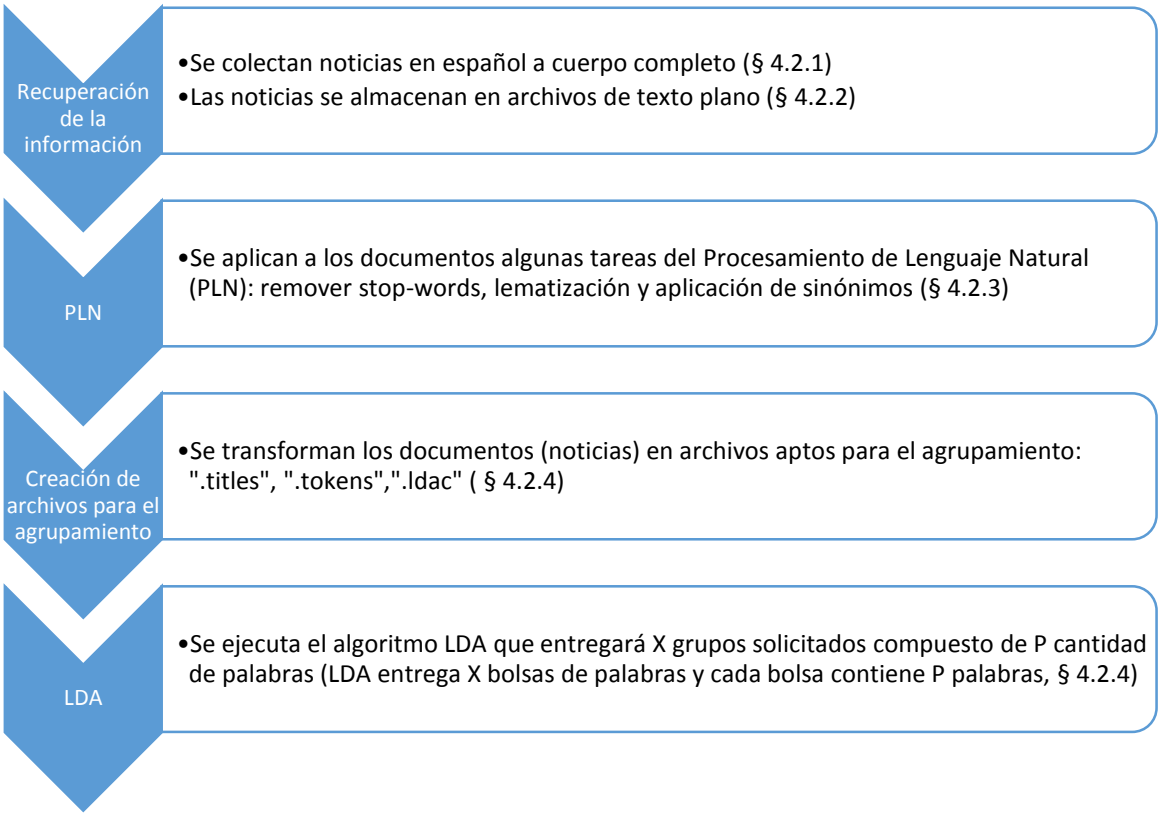
Stop Word (palabra sin sentido)
Son palabras que pueden ser ignoradas en el procesamiento orientado en escritura mediante teclado sin causar un efecto significante en la precisión. Este tipo de palabras pueden ser preposiciones, artículos, pronombres, verbos auxiliares.

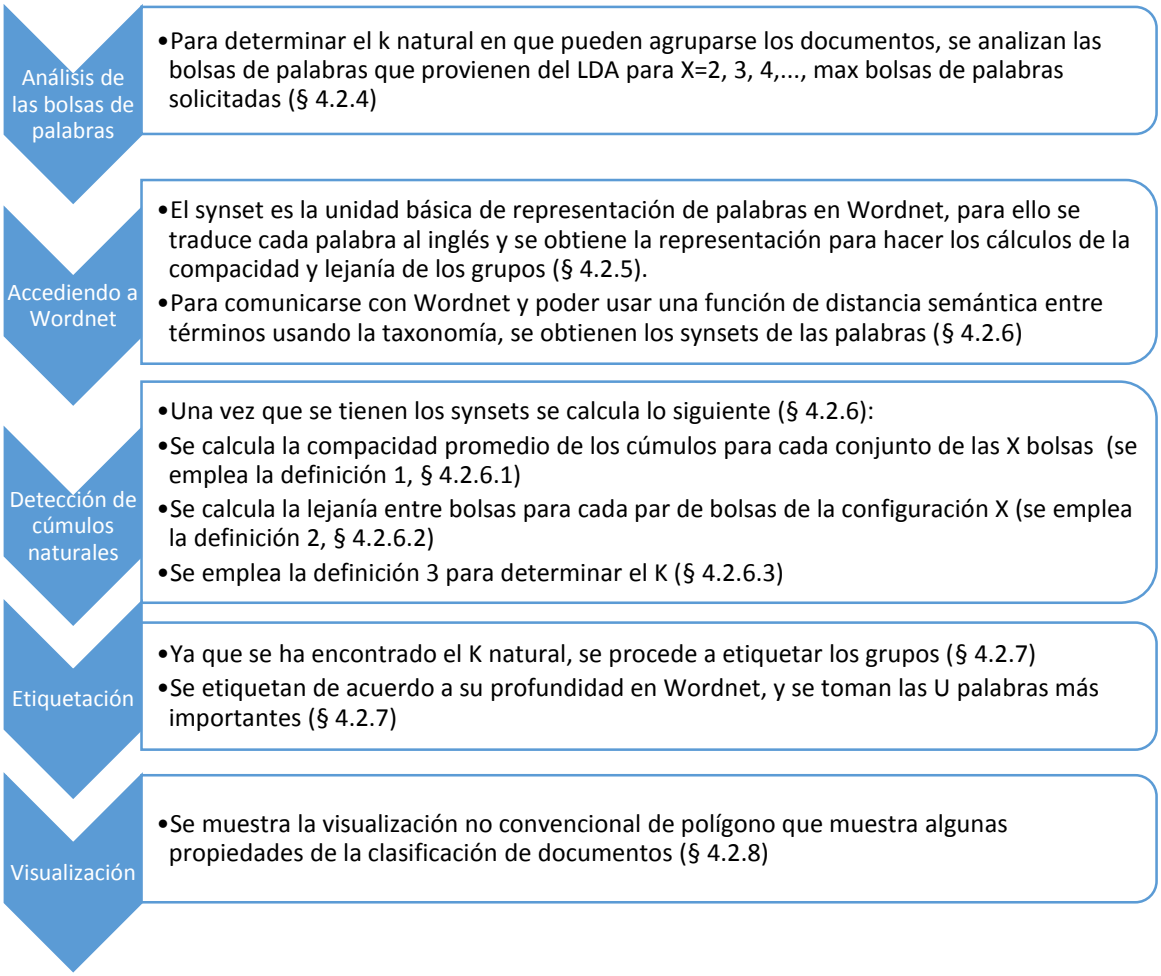
Synset [Wordnet Web]
En la jerarquía de Wordnet, los sustantivos, verbos, adjetivos y adverbios son agrupados en conjuntos de sinónimos llamados synsets, cada uno representa un concepto diferente. Los synsets se relacionan a través de relaciones léxicas y semánticas-conceptuales.

Tópico
Asunto o materia sobre la que se trata en una conversación, un discurso, un escrito, una obra artística u otra cosa semejante.

Resumen ejecutivo del modelo

Esta tesis indica cómo clasificar automáticamente documentos, en una forma que tenga sentido para los seres humanos. Cómo es de esperarse, no se conoce previamente el contenido de los documentos; no se sabe el nombre o clase a la que pertenecen; no se conoce la magnitud del conjunto de datos, tampoco se depende de almacenamiento estructurado. Se emplea la relación semántica entre palabras para sugerir el número de agrupaciones naturales.
Básicamente, el proyecto brinda tres aportaciones: la primera es, el cálculo de cúmulos naturales, para ello se relacionan dos medidas (definidas en § 4.2.6), una es la compacidad de las agrupaciones y la otra es la lejanía semántica entre grupos; ambas medidas reflejan la distancia semántica que se obtiene de Wordnet. En segundo lugar, se presenta una etiqueta que indica el contenido de los grupos, para ello se considera la importancia de los términos en la taxonomía de Wordnet. Finalmente, se presenta la clasificación de los documentos en una visualización que resalta múltiples parámetros de la clasificación (descritos en § 4.2.8).
A continuación, para favorecer la comprensión del modelo, se presenta un diagrama de lista con las etapas del mismo.




Análisis de las bolsas de palabras
•Para determinar el k natural en que pueden agruparse los documentos, se analizan las bolsas de palabras que provienen del LDA para X=2, 3, 4,..., max bolsas de palabras solicitadas (§ 4.2.4)
Accediendo a Wordnet
•El synset es la unidad básica de representación de palabras en Wordnet, para ello se traduce cada palabra al inglés y se obtiene la representación para hacer los cálculos de la compacidad y lejanía de los grupos (§ 4.2.5).
•Para comunicarse con Wordnet y poder usar una función de distancia semántica entre términos usando la taxonomía, se obtienen los synsets de las palabras (§ 4.2.6)
Detección de cúmulos naturales
•Una vez que se tienen los synsets se calcula lo siguiente (§ 4.2.6):
•Se calcula la compacidad promedio de los cúmulos para cada conjunto de las X bolsas (se emplea la definición 1, § 4.2.6.1)
•Se calcula la lejanía entre bolsas para cada par de bolsas de la configuración X (se emplea la definición 2, § 4.2.6.2)
•Se emplea la definición 3 para determinar el K (§ 4.2.6.3)
Etiquetación
•Ya que se ha encontrado el K natural, se procede a etiquetar los grupos (§ 4.2.7)
•Se etiquetan de acuerdo a su profundidad en Wordnet, y se toman las U palabras más importantes (§ 4.2.7)
Visualización
•Se muestra la visualización no convencional de polígono que muestra algunas propiedades de la clasificación de documentos (§ 4.2.8)

1. Introducción

En este capítulo se describe de forma general el panorama de la tesis, se asientan los objetivos particulares al igual que el primordial y al final del mismo se conocerá la estructura total del presente trabajo.

1.1 Planteamiento del problema

La sociedad de la información (cómo actualmente se denomina a la humanidad) [45] requiere analizar gran cantidad de datos para tomar decisiones y realizar sus actividades cotidianas, y la forma preferida para informarse por millones de usuarios a nivel global es el internet.

Según datos de [45] la proporción de la población mundial cubierta por las redes móviles y celulares es alrededor de 95%. Las redes sociales son parte importante de este impresionante tráfico de datos; en la red social Twitter se generan más 500 millones de tweets cada día [4]. Los espacios en los que se generan noticias en la Web cómo periódicos o revistas digitales son ampliamente concurridos lo cual indica la importancia de estos lugares virtuales. Las noticias también son una fuente importante de información ya que permiten conocer cuáles son los hechos más relevantes en la sociedad.

Saber cuáles son los temas que más se habla en el momento actual puede ser de gran relevancia para conocer la audiencia de ciertos programas y su reacción ante cierto fenómeno, comprender y predecir tendencias y mediar la difusión de ideas. También, pueden emplearse para encontrar cuales son los temas con mayor presencia en conjuntos de datos; por ejemplo, en el sector salud [5]. Incluso, saber los temas con mayor presencia en un conjunto de documentos con información puede ayudar a crear productos en negocios.

No obstante, una problemática común de la identificación de tópicos en fuentes de información abierta, es saber cuál es el número correcto de tópicos en los datos analizados. Además de ello, también resulta importante saber de la existencia de más de una configuración de tópicos distinguibles en la información. Sin embargo,

hoy en día ya no es suficiente agrupar las palabras de acuerdo con probabilidades de ocurrencias de las palabras; se tiene que ir más allá y emplearse análisis en la semántica de los documentos [1]; para ello, es importante contar con estructuras que permitan coordinar el análisis de la semántica entre conceptos; como un tesauro. Además, resultaría provechoso emplear algún recurso de visualización que permita representar en un mismo espacio múltiples características de la agrupación de las palabras de los documentos en cúmulos

Generalmente, se usa un clasificador supervisado para asignar clases a los documentos, no obstante, se debe saber el número de clases y enseñarle a este tipo de algoritmos como clasificar la información de manera “explícita”.

Cuando no se quieren distinguir las clases, no se saben o no se pueden etiquetar los datos es necesario usar un algoritmo de agrupamiento no supervisado (“Clustering”, en inglés). Se le dice al algoritmo que agrupe documentos de tal manera que «se parezcan entre si»; el algoritmo que se emplea dice claro que sí, pero ¿cuántos grupos quieres? Ahí se tiene el problema, no se sabe cuántas clases hay en la información, ni que tan similares son, entonces el usuario da un número aleatorio “adivinándolo” quizá, y se conforma con el resultado.

Sería de gran interés que un clasificador no supervisado hiciera un trabajo adicional y agrupara los documentos en dos, tres, cuatro, x grupos y dijera los grupos que más se parecen son estos K por tal razón; en esta tesis esos grupos estarán en función de su parecido semántico.

La solución que se presenta en este trabajo de tesis está en función explorar y analizar el comportamiento del algoritmo LDA y así como encontrar el número natural de tópicos que se encuentra en un conjunto de datos con el cálculo de intra e inter-distancias apoyándose en una jerarquía de hiperónimos e hipónimos [Wordnet], empleando visualizaciones que permitan tener información con mayores detalles. Para ello, se emplean técnicas de procesamiento de lenguaje natural para facilitar el trabajo con la semántica contenida en la información al igual que métodos matemáticos que apoyen a los algoritmos de inteligencia artificial.

1.2 Objetivos

A continuación, se presentan el objetivo general y los objetivos particulares de la tesis.

1.2.1 General

Diseñar un modelo que permita encontrar el número de cúmulos naturales en una clasificación no supervisada de conjuntos de datos no estructurados, que los etiquete y que los presente mediante recursos de visualización.

1.2.2 Particulares

- Plantear el problema de hallar el número “natural” de cúmulos
- Estudiar el estado del arte en clasificación de documentos y métricas de similitud entre documentos
- Encontrar los tópicos con mayor presencia en documentos (información no estructurada)
- Etiquetación de los cúmulos hallados
- Medir la compacidad de los cúmulos usando la taxonomía de Wordnet
- Medir la lejanía natural entre cúmulos usando la taxonomía de Wordnet
- Emplear las compacidades y lejanías naturales de los cúmulos para sugerir un número de cúmulos k que permita distinguir un número de grupos con sentido para los seres humanos.
- Crear una visualización que muestre propiedades de agrupación de documentos como: cantidad de documentos asignados a cada cúmulo, palabras representativas de los cúmulos, relevancia de las palabras para el cúmulo.

1.3 Justificación

Actualmente existe la necesidad de conocer los temas más comunes de los cuales se hablan en las noticias y además el número natural de tópicos que se encuentran en la información puede apoyar a diferentes individuos en la sociedad como reporteros, políticos, músicos, inversionistas, analistas, investigadores, entre otros.

El ser humano gusta de imágenes para comprender la información que se presenta a su alrededor y que no es capaz de analizar por diversos factores como la velocidad de los hechos o la cantidad de información a analizar. Por ello, es oportuno ayudar al usuario a que identifique de manera natural la cantidad de temas en los documentos de manera sencilla empleando modelos de similitud entre cúmulos. Un político, tal vez requiera identificar cuáles son los temas que tienen mayor importancia para su comunidad o aquellos de los que más se platica de manera cotidiana para percibir y decidir en qué tipo de problemática incursionar primero. En al ámbito tecnológico probablemente resulte benéfico conocer cuáles son los aparatos más vendidos en la actualidad y la razón de serlo; quizá un inversionista pueda emplear su dinero en cierto tipo de productos debido a su aceptación por el público o tal vez decida no hacerlo. Quizá alguna empresa requiera saber que productos se relacionan entre sí para dar recomendaciones a los usuarios con gustos similares. Los ejemplos para visualización de tópicos importantes provenientes de un análisis con enfoque semántico son diversos, sin embargo, para los especialistas e investigadores puede no ser suficiente, ya que en puede surgir la necesidad de saber en qué momento agrupar o dejar de agrupar datos para formar cúmulos y saber que tan buenos son esos cúmulos naturales de acuerdo a su relación semántica.

Los lugares óptimos para capturar información reciente además de las redes sociales son los sitios web de organizaciones cuya labor principal sea divulgar la información como lo son instituciones sin fines de lucro y periódicos; en este proyecto no se emplean redes sociales por su limitado contenido semántico y su amplio uso de reducciones e incongruencias gramaticales. Afortunadamente, muchas páginas de periódicos en internet ofrecen contenido digital reciente ya sea en formato RSS para los que buscan un rápido acercamiento a los hechos o en forma ortodoxa, que es aproximarse al artículo completo.

Las técnicas provenientes de investigadores que trabajan con lenguaje natural y las herramientas de minería de datos ayudan a trabajar adecuadamente con can-

tidades gigantescas de datos, para un ser humano es prácticamente imposible analizar millones documentos conteniendo noticias, por lo cual un modelo automático computarizado que realice esta labor resulta funcional para disminuir el trabajo y errores que puedan desprenderse de un análisis humano. Sí se complementa el estudio de los datos con un modelo de representación visual eficiente puede lograrse que el entendimiento del público en general surja de manera más rápida y sencilla.

1.4 Aportaciones principales

Las contribuciones importantes de este trabajo son:

(a) Un modelo propuesto para convertir un clasificador supervisado en no supervisado porque se desconocen de antemano las clases que se hallarán. Con éste es posible agrupar un conjunto de documentos (noticias en español de la prensa diaria) según los temas que abordan. El algoritmo se comporta como un clasificador no supervisado porque sin intervención del usuario, a cada grupo (clase) le da una etiqueta (una a tres palabras en español), que denota el tema del que habla el grupo.

(b) Estos grupos concuerdan razonablemente con los que las personas producirían, por lo que se dice que el algoritmo *extrae* la semántica de las noticias: las agrupa por los temas principales que abordan.

(c) El algoritmo maximiza la distancia semántica entre los conceptos que pertenecen a un mismo grupo, y maximiza la distancia entre grupos distintos.

1.5 Estructura de la tesis

La tesis actual se organiza en 7 capítulos. Al comienzo de cada uno se presenta una breve introducción sobre el mismo en la cual se establecen los aspectos más resaltables para introducir al lector adecuadamente en la lectura.

En el capítulo 1 se presenta el planteamiento del problema y la justificación del mismo. Además, se describe el objetivo general y los particulares de la tesis. El capítulo 2 contiene algunos trabajos que se emplean como base para el desarrollo del trabajo al igual que aquellos que forman el sustento teórico. El capítulo 3 des-

cribe las herramientas computacionales y teóricas para el desarrollo de la tesis, entre ellos están algunos conceptos, algoritmos y programas para lograr los objetivos propuestos. La metodología e implementación empleada para la realización del proyecto son descritas en el capítulo 4. Las pruebas y resultados aparecen en el capítulo 5, y en el capítulo 6 se señalan las conclusiones, proyecciones y se proponen trabajos futuros.

2. Estado del arte

En este capítulo se presentan algunos trabajos que se emplean como base para el desarrollo de la tesis al igual que aquellos que forman el sustento teórico del proyecto.

2.1 Encontrando los temas en un documento en español [2]

En [2] se muestra la forma en que se pueden encontrar los tópicos más importantes en un documento en español partiendo de la idea de ver a la información como familias de términos en forma de árboles de conceptos, por ejemplo:

Zapato {huarache, sandalia, mocasín, bota, alpargata, pantufla}

El ejemplo anterior puede verse que un mocasín es parte del conjunto zapato, lo que significa que los mocasines son zapatos, pero que no todos los zapatos son mocasines. La misma estructura de un árbol puede mostrar acciones y verbos.

Procedimiento propuesto:

Hallar los conceptos que más aparecen – esos son los temas.

1. Contar o mirar los conceptos, no las palabras.

Mientras más extenso es el artículo, mejor resulta el método.

2. Cada palabra vota (aumenta el contador de) los conceptos que denota.
3. Habrá votos equivocados. A la larga los votos correctos prevalecerán.

En [2] también se presenta el procedimiento empleado en CLASITEX que permite generar una regla de conteo para los conceptos denotados por una o más palabras, a continuación, se presenta el algoritmo.

Se recorre con un apuntador el texto del artículo en español de izquierda a derecha.

1. (Comienzo) Observa la secuencia o cadena de cuatro palabras señalada por el apuntador.
   ¿Denota algún o algunos conceptos?

Si: Aumenta el contador de cada uno de esos conceptos. Ve al paso (5).

No: ve al paso siguiente (2).

2. Observa la secuencia o cadena de tres palabras señalada por el apuntador. ¿Denota algún o algunos conceptos?

   Si: Aumenta el contador de cada uno de esos conceptos. Ve al paso (5).

   No: ve al paso siguiente (3).

3. Ídem la secuencia de dos palabras
4. Observa la palabra denotada por el apuntador. ¿Denota algún o algunos conceptos?

   Si: Aumenta el contador de cada uno de esos conceptos. Ve al paso (5).

   No: ¿Es una palabra que no denota concepto alguno?

      Si: Ignórala. Ve al paso siguiente (5).

      No: Imprímela como no sé qué significa esta palabra

5. Mueve el apuntador a la derecha de las palabras ya analizadas.

   Repite la operación o iteración: ve al paso 1 (COMIENZO). Cuando se haya analizado todo el texto, cuéntese los conceptos y repórtese los más numerosos.

2.2 Aplicabilidad de Latent Dirichlet Allocation para búsqueda en múltiples discos [6]

En [6] se calculan las estadísticas generadas por los usuarios en un RDC (Real Data Corpus). También se usa RDC para probar la efectividad del modelo LDA encontrando documentos similares dentro de un corpus de datos; se verifica la habilidad de LDA para extraer los tópicos con mayor presencia en el corpus y por último se compara LDA con una búsqueda de una expresión regular en un dominio ruidoso.

En la construcción de él corpus de datos se usaron aproximadamente 2435 unidades de almacenamiento de 25 países distintos, en la Figura 2.1 se muestran la cantidad de archivos analizados y su extensión.

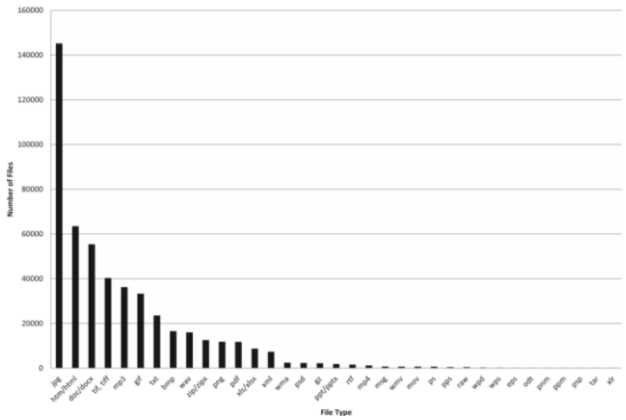


Figura 2.1: Cantidad de archivos analizados clasificados de acuerdo a su extensión. Los resultados demuestran que, mientras LDA no es un reemplazo para la búsqueda por expresión similar, si atenúa la importancia para la selección de una palabra clave de búsqueda.

El estudio concluye que documentos importantes son retornados por este modelo a pesar de no basar su funcionamiento en búsqueda por palabra clave. Se le asignó a LDA extraer documentos científicos con base en el dominio de las palabras; LDA regresó los documentos divididos en varias disciplinas. Un modelo de consulta por palabra clave tomaría más pasos para realizar la misma acción.

Por último, LDA mostró que de forma natural puede separar tópicos que se traslapan, la única limitante del modelo LDA es su velocidad de ejecución, ya que es mucho más lento que un modelo basado en búsqueda por palabra clave, sin embargo, empleando una palabra clave pueden no encontrarse los documentos importantes para el usuario debido a las limitaciones del lenguaje.

2.3 Descubrimiento no supervisado de tópicos en micro redes de blogs [7]

En [7] se presenta una metodología para descubrir los tópicos más interesantes en tweets basada en mapear hashtags en términos que entiende Wordnet y después hacer clústering apoyándose de la semántica contenida en los mensajes.

La metodología consiste en:

1. Anotación semántica

El objetivo de este paso es llenar el hueco entre hashtags y conceptos para poder comparar dos hashtags al mismo nivel semántico. Los conceptos pueden ser considerados como representaciones abstractas en clases de objetos mediante pensamientos, lenguaje y referentes. La idea de este paso es relacionar los hashtags al dominio de Wordnet y después ejecutar el proceso de clústering para aplicar medidas de similitud semántica basadas en ontologías.

2. Clústering mediante análisis de hashtag semántico

   El objetivo es agrupar todos los hashtags similares en cúmulos con términos similares para detectar tópicos de interés. Para ello se analiza la denominada matriz de similitud semántica, en la cual n representa el número total de hashtags y s_{ij} es la similitud semántica entre el i-ésimo término y el j-ésimo término calculada con una función de similitud.

$$S_n = \begin{bmatrix} S_{11} & S_{12} & S_{1n} \\ \dots & \dots & \dots \\ S_{n1} & S_{n2} & S_{nn} \end{bmatrix} |S_{ij} \mathbb{C} [0,1]|$$

Fórmula 2.1

3. Selección de tópicos

   Para simplificar la representación de los resultados, el sistema regresa una lista de tópicos y el conjunto de hashtags a cada tópico, un hashtag sólo puede pertenecer a una categoría.

En [7] se dice que la mayoría de las aproximaciones actuales de detección de tópicos en Twitter se basan únicamente en un análisis sintáctico del contenido de los tweets, apoyado por la por la frecuencia de ocurrencia de los términos.

2.4 Resumiendo los cambios en colecciones de texto dinámicas usando el modelo Latent Dirichlet Allocation [8]

En [8] se analiza la forma de resumir automáticamente los cambios en colecciones de texto con comportamiento dinámico. Se obtiene un resumen que describe los cambios más importantes hechos a un documento durante un periodo dado. Se emplea un sistema basado en LDA para encontrar los cambios escondidos en estructuras de tópicos.

La metodología consiste en:

1. Obtener artículos de Wikipedia en versión completa y formato XML
2. Se pasa de un archivo XML y se almacena en un archivo individual, cada documento recibe un formato para quedar en texto plano
3. Pre-procesamiento (filtrar vandalismo y revertir revisiones)
4. Análisis. Se comparan cuatro aproximaciones. En el primero se realiza un análisis en colecciones de texto dinámicas con un periodo definido por el usuario. Para la segunda configuración, se consideran las probabilidades conjuntas de los métodos de eliminación e inserción en el set de documentos dado un periodo. La tercera configuración se basa en el uso de LDA para encontrar cambios en la estructura de los tópicos latentes. Finalmente, la última prueba es la combinación de las anteriores, en la cual el top de las oraciones clasificadas de acuerdo con la importancia generada por el tercer método son acomodadas combinando la calificación correspondiente a la segunda aproximación.
5. Finalmente, se evalúa el desempeño de las aproximaciones usando las métricas ROUGE (Recall-Oriented Understudy for Gisting Evaluation).

El análisis de evento a través del tiempo apoya en tener una idea de todo el trabajo requerido para consultar las modificaciones que sufren los artículos para llegar a ser los documentos detallados que en muchos de los casos llegan a ser. En la Figura 2.2 pueden apreciarse los cambios sufridos por un artículo a lo largo de los años.

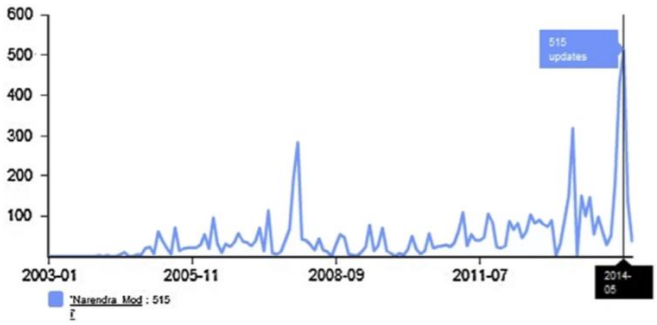


Figura 2.2: Este gráfico representa el número de ediciones sobre el tiempo en un artículo de Wikipedia denominado Narendra_Modi.

2.5 Etiquetando agrupaciones de dos perspectivas lingüística y estadística: Una aproximación híbrida [9]

La auto-etiquetación de cúmulos trata de seleccionar etiquetas apropiadas para los cúmulos que cumplan con premisas como lo son: ser fácilmente comprensibles, informativas y representativas [9].

En [9] se muestra como las etiquetas son acomodadas en una manera generativa. Además, se cuenta con que la influencia de una palabra en una calificación depende ampliamente de sus vecinos.

En la primera aproximación de esta metodología, el conocimiento lingüístico es usado para asegurarse que las frases extraídas se leen fácilmente y se comprenden. La segunda aproximación, se usa un método de calificación sensible al contexto para modelar la influencia de las palabras sobre un posicionamiento de candidatos.

Explorando la dependencia entre palabras, el artículo contribuye en:

- Proponer un algoritmo de aprendizaje de reglas el cual puede automáticamente aprender reglas de conjuntos de frases
- Proponer una regla basada en candidatos para generar posibles etiquetas
- Usar cadenas de Markov para modelar la influencia de palabras en lugar de tratarlas independientemente

- Experimentar en conjuntos de información en inglés y chino muestran que las etiquetas generadas por su modelo son fáciles de leer e informativas (Véase Figura 2.3).

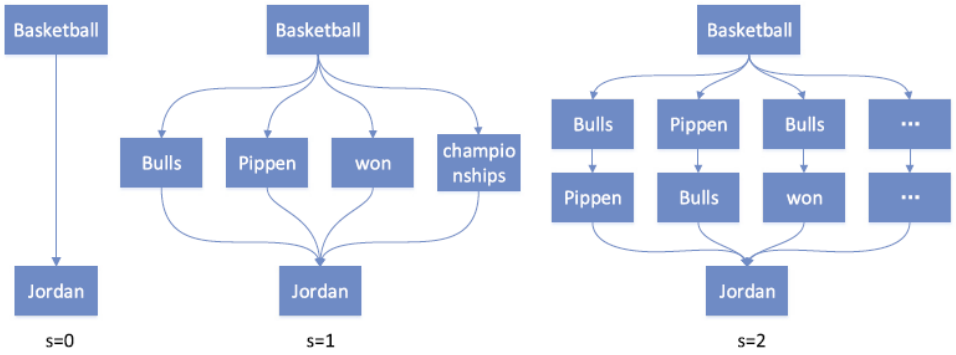


Figura 2.3: Ejemplo de etiquetación, Generando “Jordan” de la etiqueta candidato “Basketball” En [9].

2.6 Mapas auto-organizativos de crecimiento externo y aplicación en Visualización y Exploración en base de datos de E-mail [10]

En [10] se describe un modelo creado para organizar y clasificar correos electrónicos basado en mapas auto-organizativos. Para ello se emplea una red neuronal compuesta de dos capas; las neuronas de entrada corresponden a un vector de pesos. La capa de salida contiene tantas neuronas como cúmulos se necesiten.

Para calcular la similitud entre documentos se debe calcular el peso W_{ik} del término k en el documento i empleando la frecuencia del término t_f y la frecuencia inversa del documento $idf_k = \log(N/n_k)$ en la Fórmula 2.6.1.

$$w_{ik} = \frac{tf_{ik} * log(N/n_k)}{\sqrt{\sum_{j-1}^{t}(tf_{ij})^2 \left(log(N/n_j)\right)^2}}$$

Fórmula 2.6.1

Una vez obtenido el peso se calcula la similitud de la forma:

$$S(D_i, D_j) = \sum_{k-1}^{m} W_{ik} * W_{jk}$$ Fórmula 2.6.2

La etapa de entrenamiento de la red neuronal genera un mapa con la siguiente forma:

En la imagen puede apreciarse un conjunto de hexágonos. Cada hexágono representa un conjunto de mensajes que son similares, mientras más documentos se encuentren en el hexágono, más obscuro será. Cada hexágono es representado por una palabra que representa a los documentos, la similitud que tienen figuras vecinas es mayor a la que existe entre figuras mapas alejadas.

El proceso de auto-etiquetación que se emplea consiste en elegir una palabra clave para representar una celda para describir la mayor cantidad de documentos pertenecientes a esa celda, en la Figura 2.4 puede verse un ejemplo al organizar documentos en cúmulos y etiquetarlos.

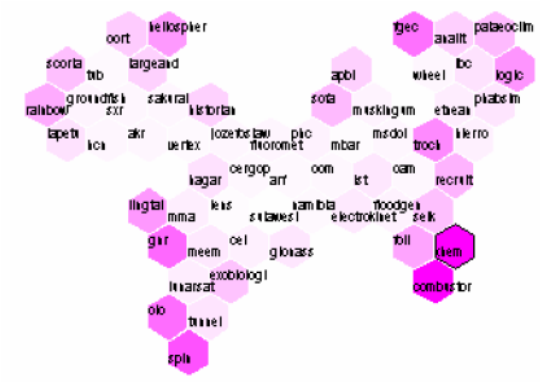


Figura 2.4: Ejemplo de la distribución de documentos etiquetados en [10].

2.7 Una aproximación semántica para agrupación de texto usando Wordnet y cadenas Léxicas [1]

En [1] se menciona que los algoritmos de agrupamiento no consideran las relaciones semánticas entre palabras, así es que no pueden representar precisamente el significado de los documentos.

Se proponen atender cuatro problemas en el artículo.

- Se propone modificar la medida de similitud basada en Wordnet para la desambiguación del sentido de las palabras. Esta se basa en la idea de que las relaciones semánticas implícitas y explícitas entre conceptos (synsets) en Wordnet imponen igualitarios factores de importancia en la medida de similitud de la palabra.
- Se emplean cadenas léxicas para capturar el tema principal de los textos. Se observó que los conceptos extraídos de cadenas léxicas son un pequeño subconjunto de las características semánticas e idealmente pueden cubrir los temas.
- El método usado puede encontrar el número de cúmulos correctos observando los resultados experimentales; útiles para k-means.
- Se demuestra que las etiquetas generadas son un buen indicador del reconocimiento y entendimiento de los cúmulos.

Se emplea el experimento de [34] para evaluar métodos para calcular la relación semántica entre palabras.

En este artículo [1], se extraen las diez palabras con mayor peso como las etiquetas para representar al cúmulo, ya que los conceptos con peso en las cadenas léxicas representativas son semánticamente importantes en términos de cúmulos. En la Tabla 2.1 se muestran etiquetas para algunos cúmulos.

Tabla 2.1: Etiquetación usando los términos con mayor peso en el experimento DCS en [1].

Group 1	Group 2	Group 3
Percentage	Uruguayan_peso	Computerized_tomography
Share	Dias	Net_income
Stock	Parnel	loss
Corporation	Delaware	Revolutions_per_minute
Parcel	Informant	Prior
Group	Compromise	Year
Million	Back	Wage
Park	Philippine	April
Stockholder	Prayer	Loss

2.8 La semántica de la confusión en jerarquías: Teoría y práctica [3]

Un ejemplo para comprender con lo que trabaja la es confusión es el siguiente: Imaginar que se tiene la pregunta ¿Qué vestir en un día lluvioso?, Un *impermeable* es la respuesta correcta. Una sombrilla tendría un error ligero en comparación con la respuesta *cinturón*; y un error grave sería decir *máquina de escribir* como respuesta. Esto es precisamente con lo que trabaja la teoría de la confusión, la medición de ese tipo de errores y la relación entre los conceptos.

2.8.1 Algunas propiedades y funciones en las jerarquías

Si se pregunta por un auto europeo y se obtiene un auto alemán, no hay error. Pero, si se pregunta por un carro alemán y sale carro europeo, ¿Se podría medir el error? Las jerarquías de valores simbólicos permiten mediar la similitud entre estos valores, y el error cuando una es usada en lugar de la otra.

Confusión usando r en lugar de s, para jerarquías simples

Si r, s ∈ H, entonces la confusión de usar r en lugar de s, escrita como conf(r,s) es:

- conf (r, r) = conf (r, s) = 0, donde s ese s cualquier superior de r.
- conf (r, s) = 1 + conf (r, father_of(s))

Confusión usando r en lugar de s, para jerarquías ordenadas

- conf'' (r, r) = conf (r, any ascendant of r) = 0;
- Si r y s son hermanos diferentes, conf'' (r, s) = 1 si el padre es un conjunto no ordenado; de lo contrario, conf'' (r, s) = la distancia relativa de r a s= el número de pasos que se necesitan para brincar de r a s en orden, dividido por la cardinalidad-1 del padre.
- conf'' (r, s) = 1 + conf''(r, father_of(s))

El conjunto de valores que son iguales a otro para un valor de confusión dado.

- Un valor u es igual a un valor v, dentro de un valor dado de confusión ε.Esto es u =εv, si y sólo si conf(u,v) ≤ε.

Similitud para valores en diferentes jerarquías y diferentes ontologías

- Cuando v_1 pertenece a una jerarquía H_1 and v_2 pertenece a otra jerarquía H_2, los dos con el mismo elemento conjunto E es mejor construir una ontología O_u de E y después medir la similitud como sigue: la similitud para dos conceoptos que pertenecen a la misma ontología sim' (cU, dU)=1/(1+longitud_path_de_ cU_hacia_du en el árbol O_u)

Similitud entre objetos y confusión acumulada

- Los objetos son entidades descritas como un set de (propiedad,valor) pares. También son llamadas relaciones-atributo.

- O´es idéntica a O si $a_i' = a_i$ para todo $1 \leq i \leq k$.
- O´es substituto de O si conf(a_i', a_i)=0 para todo $1 \leq i \leq k$.
- O´es muy similar a O si $\sum conf(a_i', a_i) = 1$.
- O´es similar a O si $\sum conf(a_i', a_i) = 2$.
- O´es algo similar a O si $\sum conf(a_i', a_i) = 3$.

En general, O´es similar$_n$ a O si $\sum conf(a_i', a_i) = n$.

2.9 Encontrando el número de cúmulos K

A continuación, se muestra una breve descripción de artículos que calcular el número de cúmulos. De acuerdo con la investigación que se realizó para encontrar los siguientes trabajos se encontró que dependiendo el tipo de datos que se ma-

nejan cuatro clasificaciones: datos numéricos, categóricos, mixtos (categóricos y numéricos) e información no estructurada (texto, para este caso).

2.9.1 Agrupación de texto basado en una estrategia divide y une [11]

En [11] se analizan características complejas incluyendo sinónimos y palabras co-ocurrentes para obtener su relación semántica. Se emplea una estrategia divide y vencerás para que la iteración converja a un número de cúmulos naturales.

Se emplea el algoritmo *K-means* para clasificar los documentos. Los métodos de agrupación son de complejidad temporal baja por lo que son buenos trabajando con conjuntos de datos grandes. Las desventajas de usar este tipo de algoritmos es que el k debe ser estimado.

A grandes rasgos, se hace lo siguiente: se extraen las características como sinónimos y palabras co-ocurrentes; los cúmulos cambian dinámicamente y se dividen o unen mediante algunas reglas; en cada iteración el centro del cúmulo y la similitud son almacenados para determinar si se dividen o unen.

Se emplea *Apriori* para extraer las palabras co-ocurrentes. Los cúmulos se inicializan de la siguiente forma: el valor inicial del punto uno se elige al azar, los siguientes se eligen con la menor similitud posible hasta completar los n-1 cúmulos restantes. Para saber la convergencia se emplea la medida F; cuando el valor de la medida F es máximo entonces se converge, es decir, es el k natural.

2.9.2 Agrupación basada la propagación de Transferencia de afinidad [12]

En [12] se establece un mecanismo agrupamiento de transferencia y es efectivo en ausencia de datos. El algoritmo propuesto *TAP* hace uso de las muestras y preferencias obtenidas de dominio como el proceso que guía al proceso de agrupación al dominio de un blanco.

TAP no requiere un número específico de k cúmulos; él toma un número real S(k,k) para cada punto k como la entada tal que los puntos en la información con valores más grandes son más probables a ser elegidos. *TAP* se usa para valores

numéricos. Se usan dos reglas actualizadas para la propagación de mensajes "responsabilidad" y "disponibilidad" para hacer uso de las propiedades geométricas obtenidas en un dominio fuente con información insuficiente.

2.9.3 Determinando el número de cúmulos usando la entropía de la información para datos mixtos [13]

En [13] se encargan de encontrar el número de cúmulos en un conjunto de datos mixtos. Emplea una variación de un algoritmo llamado *k-prototypes* el cual es una combinación se *k-means* y *k-modes* el cuál es eficiente trabajando con conjuntos de datos grandes y mixtos. Básicamente lo que hace es encontrar los cúmulos para diferentes configuraciones de K, se aplica un método denominado *cluster validity index*. Se identifica el peor cúmulo de la agrupación; se asigna cada valor x a un cúmulo de acuerdo con la medida me mínima disimilitud. Finalmente, se compara la validez de los índices y se elige un K usando una función de maximización para encontrar el pico más pronunciado en la clasificación y ese valor máximo es el K buscado.

2.9.4 Agrupación basada en el algoritmo Fuzzy K-means [14]

Datos numéricos. Se presenta un algoritmo de agrupación no supervisada aglomerativo difuso *K-means*. Es similar al algoritmo difuso de *K-means* con la distinción de que se introduce penalización por término a la función objetivo para no hacer sensible el proceso de agrupación a los centroides iniciales. El número inicial de cúmulos se fija para que se más grande que el verdadero número de cúmulos. Así, con la función de costo de la entropía cada uno de los cúmulos iniciales se moverá a los centros de las agrupaciones más densos.

2.9.5 Encontrando el número de agrupaciones en un conjunto de datos: Una aproximación teórica [15]

Se presenta un simple pero poderoso método no paramétrico para elegir el número de cúmulos basado en distorsión, una cantidad que mide el promedio de las distancias, por dimensión, entra cada observación y el centro del cúmulo más cercano. Usa datos numéricos.

2.9.6 Encontrando el número de agrupaciones en un conjunto de datos usando un algoritmo jerárquico teórico [16]

Determina el número de cúmulos usando teoría de la información y un algoritmo de agrupamiento no supervisado jerárquico top-down. El algoritmo comienza con un número muy grande de grupos y los reduce en cada iteración el número de grupos eliminando al “peor”; finalmente, midiendo el potencial de la información el número exacto de cúmulos es detectado. Este método quita sensibilidad a la inicialización de los centroides usando métodos kernel.

2.9.7 El mejor K para agrupación en datos categóricos basado en la entropía [17]

Datos categóricos. Se investiga la propiedad de la entropía de datos categóricos y propone *Bkplot* un método para determinar el un conjunto de los “mejores K “; esta aproximación se implementa con un algoritmo agrupamiento no supervisado jerárquico ACE.

2.9.8 Determinando el mejor K para agrupar conjuntos de datos transaccionales: Una aproximación de cubrimiento basada en la densidad[18]

Se propone “*transactional cluster-modes Dissimilarity”* basado en el concepto de densidad de cubrimiento como una medida de desigualdad transaccional intercúmulo. Se propone un algoritmo de clasificación no supervisada jerárquico aglomerativo demuestra buenos resultados en conjuntos de datos transaccionales.

2.9.9 Un método de inicialización para simultáneamente encontrar los grupos iniciales y el número de agrupaciones para datos categóricos [19]

Datos categóricos. Se propone una novedosa inicialización para datos categóricos el cual es implementado con el algoritmo *k-modes*. La proposición obtiene buenos centros en los cúmulos y provee un criterio para encontrar candidatos al número de cúmulos; se una un criterio de alta densidad y la distancia entre otras agrupaciones.

2.9.10 Eligiendo el número de agrupaciones [46]

La mayoría de las aproximaciones para encontrar el número de particiones se mueven alrededor de los siguientes criterios.

1. Mezcla de distribuciones

   1.1 Pruebas de hipótesis para definir entre diferentes hipótesis en el número de agrupaciones

   1.2 Constantes adicionales para para la maximización de la vecindad

   1.3 Estadísticas colaterales que deben ser mínimas en el número K correcto.

   1.4 Ajuste del número de cúmulos mientras se ajusta el modelo incrementalmente.

2. Jerarquía binaria

   2.1 Parar el criterio de división y unión con base en un criterio.

   2.2 Elegir el nivel de corte en una jerarquía completa

3. Partición

   3.1 Basados en la varianza

   3.2 Cohesión Dentro de los cúmulos contra separación entre cúmulos

   3.3 Combinando múltiples cúmulos

   3.4 Re-muestreo

2.9.11 Modelo de selección de algoritmos basado en FCM para determinar el número de agrupaciones [20]

Datos numéricos. Se propone un efectivo algoritmo difuso para determinar el número de cúmulos apoyando así a algoritmos como "*C-means"* (FCM). El número de agrupaciones se encuentra con un nuevo índice que valida los resultados de la clasificación; este índice está en función de los centroides y las vecindades.

2.9.12 Una técnica de agrupación simétrica basada en agrupación multi-objetivo para la evolución automática de agrupaciones [21]

Se trabaja con la agrupación automática de conjuntos de datos como un problema de optimización multi-objetivo. Se guardan diferentes números de cúmulos en una

cadena. Los puntos son asignados a diferentes cúmulos basándose en la simetría con respecto a un punto.

En la Tabla 2.2 se muestra una generalización de los artículos relacionados a encontrar el número de cúmulos en conjuntos de datos.

Tabla 2.2: Descripción general de artículos en los que se encuentra el número de cúmulos.

Nombre del artículo	Tipo de datos	Generalidades
Agrupación de texto basado en una estrategia divide y une [11]	Numéricos	Se extraen características semánticas de palabras, se dividen o agrupan cúmulos con base en reglas definidas. El K se encuentra empleando el valor máximo de la medida F.
Agrupación basada la propagación de Transferencia de afinidad [12]	Numéricos	Se propone TAP que no requiere un número específico de k cúmulos. Toma un número real de una función S(k,k) para cada punto k como la entada tal que los puntos en la información con valores más grandes son más probables a ser elegidos.
Determinando el número de cúmulos usando la entropía de la información para datos mixtos [13].	Mixtos	Utiliza una variación de K-prototypes. Encuentra agrupaciones para diferentes números de configuraciones de K, se comparar los índices y se elige el pico más pronunciado en la clasificación.
Agrupación Aglomerativa basada en el algoritmo Fuzzy K-means [14].	Numéricos	Similar a K-means sólo que no es sensible al proceso me inicialización de los cúmulos. Determina el K encontrando la mayor densidad en los centroides.
Encontrando el número de agrupaciones en un conjunto de datos: Una aproximación teórica [15].	Numéricos	Usa un método no paramétrico. El número de cúmulos se basa en la "distorsión", una cantidad que mide el promedio de las distancias por dimensión, usa cada observación y el centro del cúmulo más cercano.
Encontrando el número de agrupaciones en un conjunto de datos usando un algoritmo jerárquico teórico [16].	Numéricos	Se comienza con un número muy grande de grupos y se reducen en cada iteración; en cada iteración se elimina al “peor” grupo. El k se detecta midiendo el potencial de la información.
El mejor K para agrupación en datos categóricos basado en la entropía [17]	Categóricos	Se investiga la propiedad de la entropía de datos categóricos y propone Bkplot un método para determinar el un conjunto de los "mejores K".
Determinando el mejor K para agrupar conjuntos de da-tos transaccionales: Una aproximación de cubrimiento basada en la densidad [18].	Categóricos	Se propone “transactional cluster-modes Dissimilarity” basado en el concepto de densidad de cubrimiento como una medida de desigualdad transaccional inter-cúmulo.
Un método de inicialización para simultáneamente encontrar los grupos iniciales y el número de agrupaciones para datos categóricos [19].	Categóricos	La proposición obtiene buenos centros en los cúmulos y provee un criterio para encontrar candidatos al número de cúmulos; se usa un criterio de alta densidad y la distancia entre otras agrupaciones.

Eligiendo el número de agrupaciones [46].	*	Múltiples trabajos
Modelo de selección de algoritmos basado en FCM para determinar el número de agrupaciones [20].	Numéricos	El número de agrupaciones se encuentra con un índice que valida los resultados de la clasificación; este índice está en función de los centroides y las vecindades.
Una técnica de agrupación simétrica basada en agrupación multiobjetivo para la evolución automática de agrupaciones [21].	Mixtos	Se guardan diferentes números de cúmulos en una cadena. Los puntos son asignados a diferentes cúmulos basándose en la simetría con respecto a un punto.

A continuación, se muestran trabajos que se relacionan con la clasificación no supervisada del texto y un breve resumen de sus características (Véase Tabla 2.3).

Tabla 2.3: Resúmenes de artículos de clasificación no supervisada de texto.

Nombre del Artículo	Generalidades
Una técnica de agrupación para artículos de noticias usando WordNet [22].	Se usa una combinación de Wordnet y k - means. A pesar de que se calcula un k este no está en función de las relaciones semánticas entre palabras
Mapas auto-organizativos de crecimiento externo y aplicación en Visualización y Exploración en base de datos de E-mail [10].	Se clasifican e-mails usando mapas auto-organizativos; la red neuronal requiere como parámetro el número de neuronas de salida o el k
CDMI: Agrupación de documentos usando maximización de la discriminación de la información [23].	Se usa un algoritmo iterativo particional que genera k grupos transformando M dimensiones en k dimensiones, hay que definir k
Taxonomía del dominio aprendida del texto: El método de subsunción contra agrupación jerárquica [24].	se construyen taxonomías a partir de documentos, el problema es que no reflejan las relaciones semánticas en la jerarquía como se hace en Wordnet
Control difuso GA con una novedosa estrategia híbrida semántica para agrupación de texto [25].	Se emplea un algoritmo genético difuso que encuentra un modelo de espacio semántico entre documentos, sin embargo, hay que definir k
Agrupación de texto basado en la frecuencia del significado se frecuentes secuencias del significado de palabras [26].	Se proponen algoritmos que trabajan con las secuencias frecuentes de palabras frecuentes y frecuencias basado en el significado de los términos; no se estima k

3. Marco teórico

En este capítulo se explican de manera general los conceptos y herramientas empleados en el trabajo de tesis.

3.1 Recuperación de la información (RI)

Se le denomina recuperación de la información al proceso de buscar, explorar y descubrir información de repositorios de datos organizados para satisfacer las necesidades del usuario [27].

La RI contiene dos componentes fundamentales: recuperación y organización de la información. La recuperación de la información trata con la representación, almacenamiento, organización y acceso a los objetos. Estos objetos pueden ser documentos completos o párrafos, páginas web, imágenes, videos, sonidos, etc. [47] . Cabe destacar que al trabajar con RI es un hecho que se debe lidiar con información incompleta e imprecisa por parte de los usuarios y los documentos; es por ello que se emplean técnicas como el stemming, desambiguación, lematización, reconocimiento de combinaciones, entre otros [47].

Algunas partes del tratamiento de la RI según [47] son:

1. Análisis de la estructura y división en «tokens»
2. Remover palabras sin sentido “stop words”
3. Normalización morfológica
4. Peso de los términos

Este trabajo maneja una etapa de recuperación de información de fuentes abiertas, como lo son: noticias de cuerpo. La información detallada puede consultarse en la sección 4.2.

3.2 Procesamiento de Lenguaje Natural (PLN)

El lenguaje natural es la forma en que la mayor parte del conocimiento generado por la humanidad se ha transmitido. El conocimiento hoy en día sigue existiendo en documentos escritos, aunque, actualmente suelen representar por archivos

digitales. Sin embargo, la forma en que los seres humanos entienden el conocimiento, no es la misma forma en que lo hacen las computadoras. El campo de estudio que se encarga de investigar sobre la forma en cómo se resuelve el problema en que una computadora pueda comprender la información que se le presenta es a través del procesamiento de lenguaje natural [48].

En inteligencia artificial y lingüística computacional, la comprensión del lenguaje natural, es un sub-campo del procesamiento de lenguaje natural que trata con la comprensión textos a través de máquinas; para realizar esto, se debe analizar el procesamiento y análisis de mensaje que produce el programa de computadora [49]; en la Figura 3.1 puede verse el paradigma del aprendizaje humano-máquina.

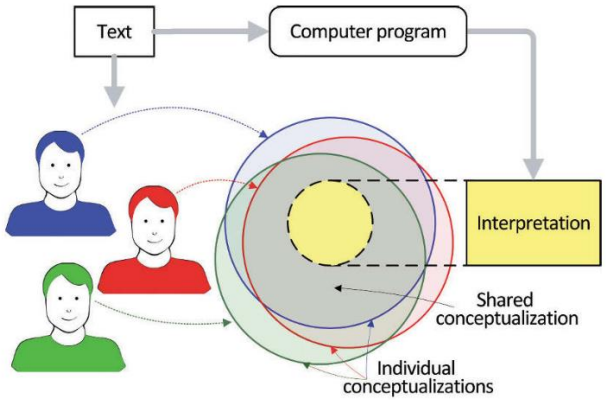


Figura 3.1: Representación del paradigma del aprendizaje humano-máquina [49].

El esquema general de la mayoría de los sistemas y métodos que involucran el procesamiento de lenguaje es el siguiente [48]:

1. El texto debe representarse formalmente conservando sus características relevantes para una tarea o método específico.
2. El programa manipula la representación transformándola según la tarea, buscando en ella subestructuras necesarias.
3. Sí es necesario, los cambios realizados en la representación formal se transforman en lenguaje natural.

3.2.1 Algunas tareas del PLN

1. Ayuda en preparación de textos
2. Búsqueda de información
3. Manejo de documentos
4. Gestión inteligente de documentos
5. Traducción automática
6. Generación de texto

3.2.2 Corpus

La cantidad de información que se puede extraer de internet para analizar es inmensa, se emplean técnicas de inteligencia artificial para analizar los datos provenientes de los corpus de información.

De acuerdo con [48], un corpus usualmente contiene marcaje especial o se prepara para facilitar la extracción de información necesaria.

3.2.3 Modelo

Un modelo es una construcción mental que refleja algunas características del objeto de investigación que son relevantes para la misma. El modelo depende totalmente del objeto que se modela [48].

3.2.4 Niveles de lenguaje

Un lenguaje natural se divide en seis niveles [48]:

1. Fonética: Exploración de las características del sonido.
2. Morfología: Estructura interna de las palabras y el sistema de categorías gramaticales de los idiomas.
3. Sintaxis: Relación de palabras dentro de la frase.
4. Semántica: El propósito de la semántica es “entender” la frase. Por lo tanto, se necesita saber el sentido de todas las palabras e interpretar las relaciones sintácticas.
5. Pragmática: Trata la relación de la oración y el mundo exterior.

6. Discurso: Normalmente se habla empleando una serie de oraciones; estas oraciones tienen cierta relación entre sí. Las relaciones hiladas forman este tipo de recurso.

3.2.4 Lematización

La lematización es la reducción de las formas flexivas de los lexemas que aparecen en un texto a su respectivo lema o forma de cita convencional [43].

Ejemplo:

El lexema de las siguientes palabras:

"Corrí", "carrera", "correr", "corredor"," correrán" y "corriste"

Es:

"correr"

La información que se analiza en el trabajo es tratada con tres actividades como lo son: lematización, remover stop-words y aplicación de sinónimos (Véase sección 4.2.3). Además, también se aplica una etapa de traducción para acceder a los recursos de Wordnet.

3.3 Minería de datos

La Minería de datos es "El descubrimiento de modelos para los datos" [50]. El proceso debe ser automático o (como usualmente lo es) semiautomático [51].

La minería de datos es una rama computacional que combina técnicas de inteligencia artificial, estadística, bases de datos, visualización y gran procesamiento de datos; puede proveer herramientas para descubrir conocimiento de los datos [Jiawei Han 2012].

El proceso de descubrimiento de los datos consta de las siguientes etapas [Jiawei Han 2012]:

1. Limpieza de los datos (Remover ruido e inconsistencias en la información).
2. Integración de los datos (Múltiples fuentes de información se combinan).

3. Selección de datos (Información relevante para el análisis puede ser recuperada de la base de datos).
4. Transformación de los datos (La información se transforma y consolida para aplicar las operaciones pertinentes).
5. Minería de datos (Se aplican métodos inteligentes para extraer información de los datos).
6. Evaluación de patrones (Identificar patrones realmente interesantes).
7. Representación del conocimiento (Se usan técnicas de visualización y representación del conocimiento para mostrar a los usuarios la información obtenida).

3.4 Minería de texto

Consiste en analizar grandes cantidades de texto y descubrir el conocimiento, que no está literalmente en cualquiera de los documentos. Es un área emergente, que combina la minería de datos y el procesamiento de textos (grandes) [52].

3.4.1 Actividades fundamentales de la minería de texto:

1. Recuperación de la información
2. Extracción de la información que puede ser obtenida

Se destaca que este trabajo ocupa la minería de datos para conocer los tópicos que pueden formarse con la información de entrada (noticias) a través de un algoritmo de inteligencia artificial denominado LDA (Véase sección 3.7).

3.5 Modelo de Espacio Vectorial (MEV)

Para analizar grandes cantidades de texto mediante computadoras deben aplicarse técnicas de aprendizaje automático, para realizar esto debe emplearse un modelo de representación de los documentos el más común y probablemente el único método es el *Modelo de Espacio Vectorial* [28].

El Modelo de Espacio Vectorial es ampliamente usado en las ciencias de la computación debido a su simplicidad y su base conceptual corresponde a la intuición del humano al procesar información.

Para construir este tipo de representación (MEV) se requiere la selección de características en los datos, este proceso es subjetivo, aunque el proceso de comparación entre vectores es puramente objetivo.
El modelo se basa en representar un documento por n dimensiones o atributos. Estas características son usadas para describir las características de los documentos [27].

$$d_i = (a_{i1}, a_{i2}, \dots, a_{ij}, \dots, a_{in}) \qquad \text{Fórmula 3.5.1}$$

En la Fórmula 3.5.1, d_i es un documento, a_{ij} es una característica que describe al documento, el valor de sus pesos refleja la importancia de la característica a_{ij} en el documento d_i, los valores posibles para el peso van de 0 hasta el infinito.

3.5.1 Matriz término documento

Una matriz de "término documento" consiste en un grupo de vectores documento, las filas son documentos y las columnas son características respectivamente.

$$D = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & a_{ij} & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \qquad \text{Fórmula 3.5.2}$$

En D (Véase Fórmula 3.5.2), a_{ij} es el peso del documento d_i para el rasgo j, m es el número de documentos en la colección.

Algunas fortalezas de usar el Modelo de Espacio Vectorial [27] son:

1. El MEV es apropiado para representar un objeto con múltiples características de forma natural.
2. Los pesos pueden ser ocupados para relacionar la importancia de un término en el documento, permitiendo a los términos ser más o menos importantes dentro de un documento.

3. Basado en el MEV, gran variedad de métodos de cálculo para medir la similitud entre objetos puede ser desarrollados como, basados en distancias o basados en ángulos.
4. La naturaleza iterativa de la recuperación de la información provee mecanismos para el ajuste de estrategias de búsqueda de manera dinámica.
5. Un Modelo de Espacio Vectorial provee un ambiente ideal donde sofisticadas técnicas y métodos como mapas auto-organizativos, modelos escalables multidimensionales, modelos basados en distancias y ángulos, etc. Pueden ser desarrollados e implementados.

En esta tesis el modelo de espacio vectorial se empleó para generar uno de las entradas del algoritmo LDA (Véase sección 4.2.4).

3.6 Reconocimiento de Patrones

Las tareas básicas del área conocida como “reconocimiento de patrones” son cuatro: clasificación, regresión, recuperación y agrupamiento.

3.6.1 Clasificación

El problema de la clasificación consiste en tomar vectores de entrada o patrones y decidir a cuál de las N clases pertenecen, para ello se entrena con ejemplares de cada clase [53]. Para realizar la asignación se emplea una estructura conocida como clasificador.

La clasificación puede emplear cualquiera de los dos paradigmas siguientes:

- Supervisado

En el paradigma supervisado, un maestro provee una etiqueta de categoría o costo para cada patrón del conjunto de entrenamiento [54].

- No supervisado

En el enfoque no supervisado, no existe un maestro explícitamente; el sistema forma cúmulos o “grupos naturales” con los patrones de entrada [54].

En esta tesis, se presenta una metodología en la que un clasificador no supervisado puede comportarse como uno supervisado, es decir, se parte de un algoritmo

no supervisado como lo es LDA y analizando la semántica de los cúmulos obtenidos por el programa puede saberse naturalmente las agrupaciones semánticamente más relacionadas.

3.7 Latent Dirichlet Allocation (LDA)

Es un modelo generativo probabilístico comúnmente empleado para identificar tópicos latentes dentro de un conjunto de datos [6] en colecciones de documentos de texto [5], no está únicamente ligado a trabajar con el mismo. De acuerdo con [6] LDA se emplea para proveer una técnica útil que ayuda a filtrar material ruidoso y aislado.

Cada tópico en turno es como una mezcla infinita de probabilidades de cada tópico; en el contexto de modelado de texto las probabilidades del tópico proveen una representación explícita del documento [5].

3.7.1 Generalidades LDA

- Una palabra es la unidad básica de los datos, la forma de representar las palabras es a través de vectores que tienen una única componente.
- Un documento es la secuencia de N palabras denotado por una tupla de la forma $\boldsymbol{w} = (w_1, w_2, \dots, w_N)$.
- Un corpus es una colección de M documentos denotado por $\boldsymbol{D} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_M)$.
- α parámetro de Dirichlet sobre las distribuciones de tópicos por documento.
- β parámetro de Dirichlet, previo a la distribución de palabra por tópico.

Dados los parámetros a y b, la probabilidad conjunta de una serie de tópicos q, un conjunto de N tópicos **z**, y un conjunto de N palabras **w** está dado por:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(Z_n|\theta)\, p(w_n|Z_n, \beta)$$

Fórmula 3.7.1

Donde $p(z_n|\theta)$ es θ_i para un único i tal que $z_n^i = 1$. Integrando sobre θ y sumando sobre z, la distribución marginal de un documento es:

$$p(\mathbf{w}|\alpha,\beta) = \int p(\theta|\alpha)\left(\prod_{n=1}^{N}\sum_{Z_n} p(Z_n|\theta)p(W_n|Z_n,\beta)\right)d\theta$$

Fórmula 3.7.2

Finalmente, multiplicando las probabilidades marginales de los documentos individualmente, se sabe que la probabilidad de un corpus es:

$$p(D|\alpha,\beta) = \prod_{d=1}^{M}\int p(\theta_d|\alpha)\left(\prod_{n=1}^{N_d}\sum_{Z_{dn}} p(Z_{dn}|\theta_d)p(W_{dn}|Z_{dn},\beta)\right)d\theta_d$$

Fórmula 3.7.3

El modelo LDA se representa gráficamente en la Figura 3.2.

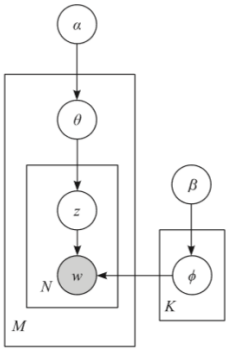


Figura 3.2: Modelo gráfico representando Latent Dirichlet Allocation. Las cajas son “platos” representando réplicas. El plato externo representa documentos, mientras el plato interno representa la acción repetida de tópicos dentro un documento.

LDA es el algoritmo que con el que se realizan las agrupaciones de los documentos de entrada y se extraen las palabras que representan a cada cúmulo; estas palabras se emplean para calcular las intra-distancias e inter-distancias (Véanse secciones 4.2.4 y 4.2.7).

3.8 Modelo de similitud y distancia entre tópicos

En [55] se presenta un modelo formal para comparar tópicos. Se basa en hacer una serie de operaciones tomando como factor fundamental la distancia entre la posición de las palabras que representan los tópicos. A continuación, se explican las componentes de función distancia creada en [55].

La distancia que existe entre dos tópicos que se encuentran en diferentes periodos P^i y P^j con *i* ≠ *j*, donde *i* y *j* van desde 1 a *P* se puede expresar de la siguiente manera:

$$T_l^i = \{W_l^i 0, W_l^i 1, W_l^i 2, \dots, W_l^i n_i\}$$
$$T_h^j = \{W_h^j 0, W_h^j 1, W_h^j 2, \dots, W_h^j n_j\}$$

Donde:

- ***i*** y ***j*** son periodos distintos desde 0 a P.
- ***l*** y ***h*** son los tópicos que se encuentren en los periodos *i* y *j* respectivamente.

T_l^i y T_h^i pertenecen al conjunto de tópicos en los periodos P^i y P^j respectivamente.

W_l^i Es el conjunto de palabras que se encuentran en el tópico $T_l$del periodo P^i.

W_h^j Es el conjunto de palabras que se encuentran en el tópico T_h del periodoP^j.

La distancia se define como sigue:

$$Dist(T_l^i, T_h^j) = \left[\frac{2}{N(N+1)}\right] * \sum_{n=1}^{N} |pos(w_l^i n) - pos(w_h^j n)|$$ Fórmula 3.8.1

Observación. Si $w \in T_l^i$ y w $\notin T_h^j$ su posición en ***h***, ***pos (w)*** es 0.

La función ***pos (w)*** determina el valor de la posición de *w* dentro de los tópicos *l* y *h*.

Si la $Dist(T_l^i, T_h^j) = 0$, los tópicos son los mismos.

Si la $Dist(T_l^i, T_h^j) = 1$, los tópicos son completamente diferentes, no tienen nada en común.

Si la $0 \leq Dist(T_l^i, T_h^j) \leq 1$ los tópicos tienen algunas palabras en común.

Observación. Cuando se compara un tópico con varios tópicos de otros periodos, se selecciona el tópico con menor distancia:

$$Dist(T_l^i, T_h^j) \text{ y } Dist(T_l^i, T_m^j)$$

$$Dist(T_l^i, T_h^j) = min\left[\left[\frac{2}{N(N+1)}\right] * \sum_{n=0}^{N} |pos(w_l^i n) - pos(w_h^j n)|\right] \quad \text{Fórmula 3.8.2}$$

3.9 Visualización de la información

De acuerdo con [27], “visualización” es el proceso de transformar datos, información y conocimiento en representaciones gráficas para apoyar tareas como análisis de datos, exploración de la información, predicción de tendencias, detección de patrones, entre otras.

3.9.1 Clasificación de las visualizaciones

La visualización se divide en dos grandes ramas:

1. Visualización científica

Cotidianamente se emplea para extender la percepción se los sentidos humanos mostrando escalar en escalas que el ojo humano pueda captar [27]. Algunos ejemplos de este tipo de visualización son: forma de las moléculas, dinámica de fluidos, astrofísica, y muchas más.

2. Visualización de la información

La visualización de la información se emplea para apreciar información abstracta [27]. Algunos ejemplos son: razonamiento visual, modelado visual de datos, programación visual, razonamiento espacial y visualización de sistemas.

La visualización de la información no posee una estructura fija, es decir, se adapta al contenido que representa. Mientras que la estructura de la visualización científi-

ca posee una estructura definida para ilustrar los datos. Resta por adherir que el hecho de que la visualización de la información no posea una estructura definida da cabida a nuevos modelos desarrollados con la imaginación en una forma diferente a la de sólo dibujar objetos simples.

Algunos modelos de visualizaciones pueden ser los siguientes: modelo basado en métricas difusas [29]; empleando mapas auto organizativos [10]; o visualización para recuperación de la información [30].

La visualización desarrollada en este trabajo se emplea para destacar varias propiedades de la clasificación (Véase sección 4.2.8). Se parte del enfoque de la visualización de la información descrito en la sección 3.9.1.

3.10 Herramientas computacionales

3.10.1 Anaconda

Es una distribución gratuita de Python que contiene 300 de los más populares paquetes de Python para ciencias, matemáticas, ingeniería, análisis de datos. Entre los paquetes más destacados se encuentran Numpy, NLTK, SciPy, MatplotLib, Scikit-learn, Qt, Pandas.

3.10.2 Pycharm

IDE inteligente de trabajo para Python. Gracias al uso de esta herramienta la ejecución del código en Python se hace más sencilla ya que la herramienta se encarga de instalar los paquetes faltantes al momento de compilar los scripts, permite el uso sencillo y eficiente de Python.

Algunas características del entorno son: terminal embebida; control de versiones; soporte para integración sencilla de plug-ins; prueba de errores (Debugger); análisis de código; auto-identación y búsqueda rápida de documentación

Este IDE es de paga, aunque se tiene una versión “Community” para labores sencillas completamente gratuita. Afortunadamente, la compañía ofrece un año en versión completa para estudiantes. Generalmente en Windows es caótico trabajar con módulos de Python por la laboriosa instalación requerida; se seleccionó a Pycharm ya que permite elegir la versión de Python con la cual se desea trabajar.

Además, es posible instalar módulos y dependencias sin requerir configuraciones profundas del sistema.

3.10.3 Software basado en Python

En esta sección se mencionan las librerías empleadas en el proyecto. Todas vienen incluidas en Anaconda.

3.10.3.1 Lda 1.0.3

Modelador de tópicos con Latent Dirichlet Allocation; emplea muestreo de Gibbs colapsado.

3.10.3.2 Numpy

Es un paquete para cómputo científico con Python. Entre sus principales ventajas están, una capacidad enorme para trabajar con arreglos de n dimensiones; sofisticadas funciones de comunicación; herramientas para integrar código c/c++ o Fortran; uso de álgebra lineal; transformada de Fourier y capacidad para trabajar con números aleatorios.

3.10.3.3 SciPy

Es un ecosistema basado en Python de código abierto para matemáticas, ingeniería y ciencia.

3.10.4 3.10.6 Freeling

3.10.4.1 ¿Qué es Freeling? [31]

Freeling es una suite de código abierto que provee servicios de análisis de lenguaje. Sí se requiere un desarrollo, por ejemplo, un sistema de traducción máquina, y se requiere algún tipo de procesamiento lingüístico se puede llamar a los módulos de Freeling.

3.10.4.2 ¿Qué no es Freeling? [31]

Freeling no es una herramienta de análisis de texto orientada hacia el usuario. No se diseñó para ser amigable con el usuario. Los resultados arrojados por la librería análisis lingüísticos en una estructura de datos.

Los idiomas con los que trabaja Freeling son: Austriaco (As), Catalán (ca), Inglés (en), Francés (fr), Gallego (gl), Italiano (It), Portugués (pt), Ruso (ru), Esloveno (sl), Español (sp), y Galés (cy).

3.10.4.3 Información lingüística de Freeling

En el sitio oficial de Freeling [62] puede consultarse un diccionario que apoya al procesamiento de lenguaje natural que contiene alrededor de 555 000 formas de palabras correspondiente a 76000 lemas. Este diccionario puede obtenerse de MCR (Multilingual Central Repository) [63].

La lematización que se realiza en este proyecto se realizó empleando Freeling (Véase sección 4.2.3.3).

3.11 Medidas de similitud semántica

La medición de la similitud semántica es un tema central en inteligencia artificial, psicología y ciencia cognitiva. Se ha usada ampliamente en procesamiento de lenguaje natural, recuperación de la información, desambiguación de palabras, segmentación de texto, contestación automática de preguntas, sistemas de recomendación, extracción de la información [32].

En general, las medidas pueden ser divididas en cuatro ramas [1]: las basadas en longitud o trayectorias; basadas en el contenido de su información; basadas en sus características y las híbridas [32].

A continuación, se da una lista de las funciones de similitud en Wordnet.

- Medidas basadas en trayectorias

La similitud entre dos conceptos está en función de la longitud del camino que liga los conceptos con respecto a su posición en la taxonomía.

‘The shortest path based measure’, ’Wu & Palmer MEasure’, ‘Leakcock& Chodorow´s measure’ y ‘Li´s measure’

- Medidas basadas en el contenido

Mientras más información comparten un par de conceptos, más parecidos son.

‘Resnik´s measure’, ‘Lin´s measure’, ‘Jiang´s measure’

-Medidas basadas en características

Las medidas basadas en características son independientes de la taxonomía y trata de explotar las propiedades de la ontología para obtener los valores de similitud.

-Medidas híbridas

Combina las ideas presentadas arriba. En algunos casos se aplican las relaciones "es-un" y "parte de".

3.11.1 Similitud basada en el camino más corto [32]

La medida toma en consideración la longitud del término uno con respecto al segundo, longitud (c_1, c_2). La medida asume que la sim(c_1, c_2) depende de que tan cercanos están los conceptos en la taxonomía, en este caso la de Wordnet. Se basa en dos observaciones. Una es que el comportamiento de la distancia conceptual es una métrica. La segunda, es que la distancia conceptual es proporcional al número de aristas que separan los conceptos en la taxonomía.

$$sim_{path}(c_1, c_2) = 2 * deep_{max} - len(c_1, c_2) \qquad \text{Fórmula 3.11.1}$$

Véase [32] para conocer más medidas de similitud.

3.11.2 Similitud Wu y Palmer [1]

Wu y Palmer calcularon la similitud entre dos sentidos[1] encontrando el menor ancestro en común (LCS) nodo que conecta sus sentidos. Es decir, es el ancestro común más cercano a cualquiera de los dos sentidos.

Por ejemplo, puede verse de los rectángulos rojos en la Figura 3.3, el LCS de canino y compadre es organismo, porque es el nodo mínimo entre los caminos de estos dos sentidos de la raíz de la jerarquía de Wordnet, (Véase Figura 3.3).

Una vez que el LCS ha sido identificado, la distancia entre los dos sentidos se calcula como:

$$sim_{WP}(C_1, C_2) = \frac{2*depth(lso(c_1,c_2))}{len(c_1,c_2)+2*depth(lso(c_1,c_2))} \qquad \text{Fórmula 3.11.2}$$

Donde, d es la profundidad del LSO de la raíz, L_p es la longitud del camino entre C_1 y el LSO.

[1] Uso "sentido" como "sentido de un vocablo", uno de sus posibles significados. Es lo que en representación del conocimiento se conoce como concepto. No es el sentido (dirección) de un camino o trayectoria.

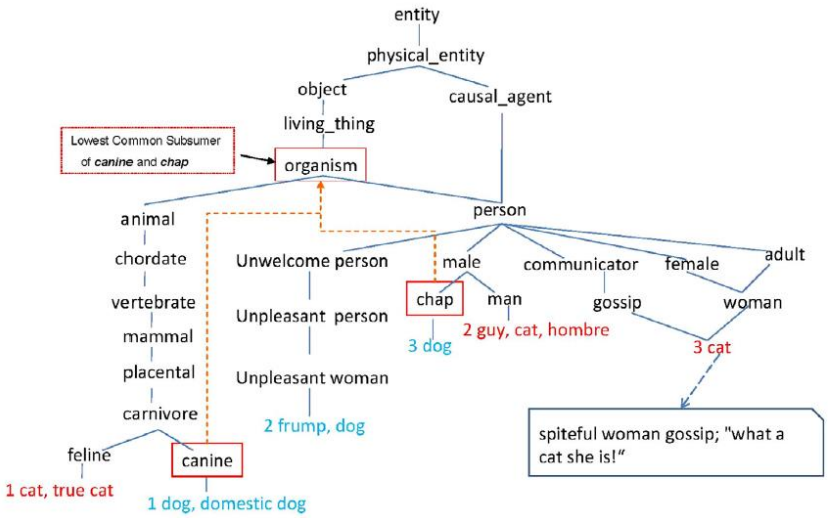


Figura 3.3: Muestra de la Jerarquía de Wordnet [1].

La etapa del cálculo de las intra e inter distancias indica la similitud para determinar la compacidad y lejanía natural entre tópicos. Para calcular estos valores, es necesario determinar la similitud entre palabras, y este valor debe basarse en su parecido semántico (Véase sección 4.2.7).

3.12 Leyes del texto

En esta sección se habla de un par de leyes que deben cumplir el comportamiento de las palabras en los documentos; estas dos son la ley de [56] y la ley de Heaps [57].

3.12.1 Ley de Zipf [Zipf 1949]

En estudios cualitativos del lenguaje, la frecuencia de cada palabra en la escritura o discurso es claramente la propiedad más elemental del lenguaje humano [33].
La ley de Zipf da una idea de cómo la frecuencia de distribución de palabras debería lucir [58]. En la Figura 3.4 se muestra el comportamiento típico de la Ley de Zipf.

Figura 3.4: Distribución de frecuencia para los términos que aparecen en una pequeña muestra de Google+ “Hugs” [58].

La formulación original establece que en una muestra escalable del lenguaje (texto o discurso) el número de palabras $Z(n)$ el cual ocurre n veces decae con n como:

$$Z(n)\ \alpha\ n^{-\delta}$$ Fórmula 3.12.1

Para un rango de valores de n. El exponente δ cambia de texto a texto, pero frecuentemente se encuentra que $\delta \sim 2$.

3.12.2 Ley de Heaps [57]

Esta ley empírica describe el número de palabras distintas, N, en un documento (o conjunto de documentos) como función de la longitud, z: el resultado clásico para esta relación es la siguiente ley:

$$N_k = N(z)\alpha K^{\gamma}$$ Fórmula 3.12.2

Con $\gamma = [0,1]$.

En la Figura 3.5 se muestra el comportamiento esperado para le Fórmula 3.12.2.

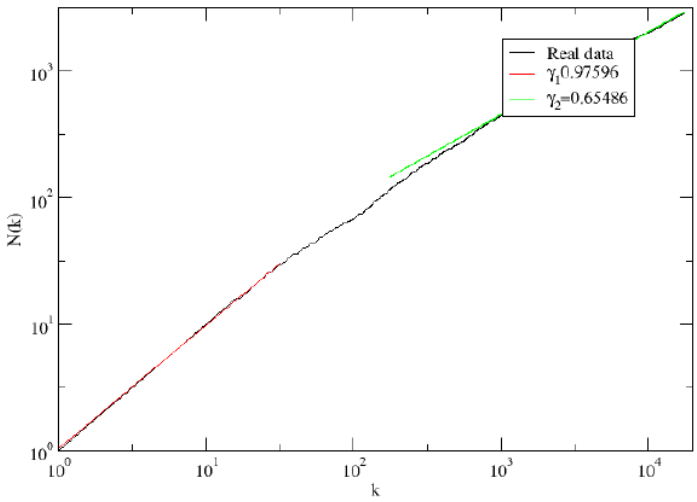


Figura 3.5: N(z) en una gráfica log-log para el libro ‘War and Peace’ en inglés.

4. Metodología e Implementación

En este capítulo se describe la metodología desarrollada.

4.1 Desarrollo del trabajo visualización de tópicos interesantes mediante un enfoque semántico

4.1.1 Arquitectura del sistema

En la Figura 4.1 puede verse un diagrama general del proyecto, posteriormente se detalla cada etapa.

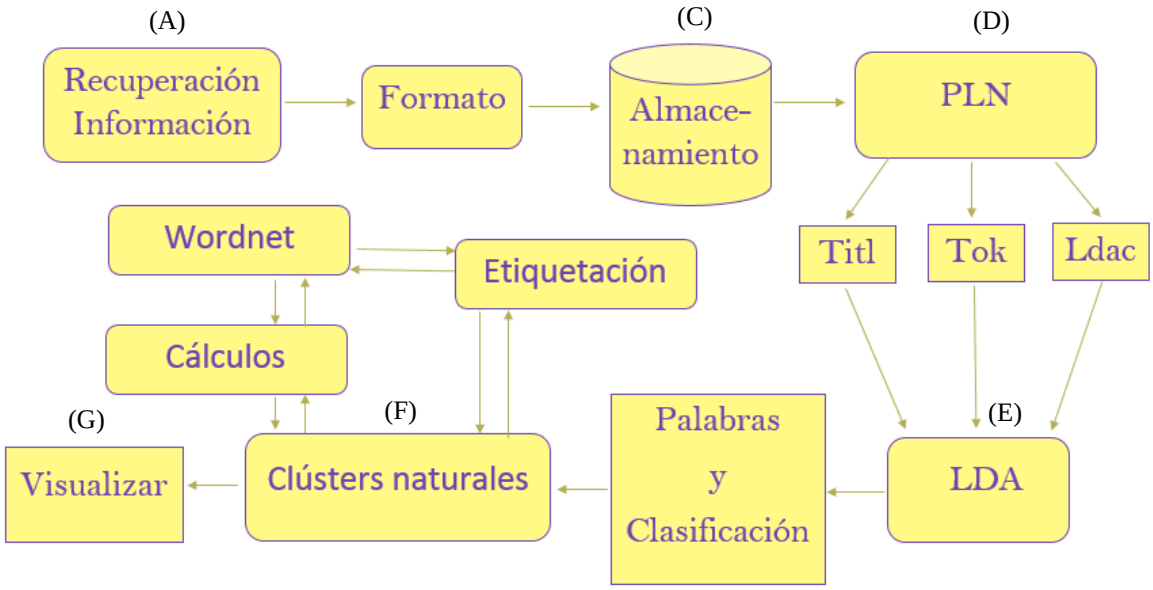


Figura 4.1: Metodología de la tesis: 'Visualización de tópicos importantes usando un enfoque semántico'. Básicamente, los documentos recuperados en (A) se guardan en (C), se procesan bajo tres técnicas en (D) y se someten al clasificador no supervisado (E). Luego viene la etapa de hallar los cúmulos naturales (F) usando Wordnet, para finalmente visualizarlos (G).

4.2 Descripción por etapas

En esta parte se detallan los bloques de la arquitectura del sistema.

4.2.1 Recuperación de la información

La información requerida por este proyecto proviene de diferentes fuentes (véase 4.2.1.1). Se recolectaron más de mil noticias almacenadas en diferentes conjuntos de datos en la Tabla 4.1 pueden verse información relacionada con los documentos.

Tabla 4.1 Conjuntos de datos empleados para probar el modelo. Columna uno, nombre; columna dos, cantidad de noticias; columna tres, cantidad promedio de palabras en las noticias.

Nombre del conjunto de datos	Número de noticias	Promedio de palabras
Volcanes	19	531
Universo	66	599
Suecia	100	421
Siria	100	484
Salud	100	501
Louvre	54	611
EPN	100	375
Salud_El_Sol	20	393
Corea	32	651
Béisbol	100	375
Tecnología (CONACYT)	21	834
Sociedad (CONACYT)	12	888
Noticias Generales	522	130
Gripe	6	1068
Ciencia(CONACYT)	46	711
Ballenas	60	315
Tortugas	42	392

Las noticias se recuperaron usando el cuerpo completo de la misma, es decir, no se emitieron datos como se hace al usar el formato RSS.

4.2.1.1 Sitios web de noticias

Para realizar la recuperación de la información se seleccionaron algunos de los periódicos más populares en México (versión en web), así como instituciones que difunden el conocimiento como lo son el CONACYT y la OMS, con la única consigna de elegir las noticias en forma completa. La recolección manual de noticas tiene ventajas como se menciona en [2] y [8].

Las noticias recuperadas manualmente no presentan desventajas de longitud o código basura como sucede al trabajar con el formato RSS o tweets, ya que se

toman uniformemente y con todo el contenido semántico de la fuente, simplemente se copia la noticia como se presenta en el sitio web sin modificaciones.

Sitios web empleados para la recolección de noticias:

1. CNN Expansión [64]
2. Reforma [65]
3. El Universal [66]
4. 20 Minutos [67]
5. Noticias de la ciencia [68]
6. BBC [69]
7. CONACYT [70]
8. OMS [71]

4.2.2 Almacenamiento

Por la simplicidad que ofrece Python al trabajar con archivos de extensión ".csv" se opta por emplear directamente a los mismos y guardar en forma sencilla aquellos conjuntos de noticias que se requieran. Además, esta forma de almacenamiento evita estar limitado a trabajar únicamente con la información en una base de datos, ya que sí se requiere analizar conjuntos de externos pueda hacerse de manera sencilla. Por último, cabe destacar que este proyecto puede ser utilizado por diferentes usuarios que no saber crear una base de datos o guardar registros en una, por ello, resulta útil trabajar con estos archivos.

No obstante, para cubrir posibles mejoras al sistema, se crea una base de datos. De las noticias recuperadas simplemente se eliminan todos los saltos de línea para que el texto quede continuo; además, se agrega una variable con la fecha de emisión. Ya que los archivos provienen de diferentes fuentes, se estandarizó la fecha de emisión de las noticias para facilitar el almacenamiento en una base de datos. Se empleó el lenguaje MySQL para contener la información; el formato a convenir fue "AAAA-MM-DD" (Se emplea una función en Python para modificar las fechas). La base de datos que almacena las noticias se nombró "BD Noticias".

4.2.3 Tratamiento lingüístico

Para este proyecto se ocuparon las siguientes tareas de procesamiento de lenguaje natural: remover palabras sin sentido (stop-words), lematización e implementación de sinónimos. Aunque no es propio del PLN y se considere como evidente, se hace mención que todos los documentos (noticias) son convertidos a minúsculas para tener uniformidad en el tratamiento de la información, recordar que la mayoría (si no es que todos) los lenguajes de programación son sensibles a la detección de mayúsculas y minúscula. Además, se removieron algunos signos de puntuación como “ ” ? ”, “|”, “!” “.

El siguiente es un ejemplo de texto sin el tratamiento de lenguaje natural empleado en la tesis después de convertir el texto en minúsculas; se tomó de [70] y lleva por título “Vivienda ecológica para zona semidesértica de Coahuila”.

“vivienda ecológica para zona semidesértica de coahuila. la facultad de arquitectura unidad saltillo de la universidad autónoma de coahuila desarrolla el proyecto vivienda ecológica para la zona semidesértica para el ejido narigua municipio de general cepeda coahuila mediante el uso de arquitectura de tierra apoyada y reforzada con tecnologías sustentables. zona semidesertica 12 “lo más importante es respetar el medio ambiente, no alterarlo más hacer una casa amigable de confort para las personas de los ejidos; con este proyecto de investigación estamos proponiendo reducir las cantidades de materiales y energía utilizada en la extracción de recursos naturales. estamos dejando de lado la arquitectura de nuestros principios, la arquitectura de tierra que está comprobada que es perdurable.”

A continuación, se explica el desarrollo de las tres tareas de PLN usadas en este proyecto.

4.2.3.1 Remover stop-words

Para remover las stop-words se diseñó un método en Python para eliminar cualquier ocurrencia de cada documento de cualquiera de las palabras contenidas en una lista denominada “stop-words”. Esta lista contiene más de 300 palabras que no son útiles al analizar texto mediante computadora, la lista es proporcionada por Google y puede descargarse de [72].

A continuación, se encuentra el texto después de remover las stop-words.

“ vivienda ecológica zona semidesértica coahuila facultad arquitectura unidad saltillo universidad autónoma coahuila desarrolla proyecto vivienda ecológica zona semidesértica ejido narigua muni-

cipio general cepeda coahuila mediante uso arquitectura tierra apoyada reforzada tecnologías sustentables zona semidesertica 12 importante es respetar medio ambiente alterarlo casa amigable confort personas ejidos proyecto investigación estamos proponiendo reducir cantidades materiales energía utilizada extracción recursos naturales estamos dejando lado arquitectura principios arquitectura tierra está comprobada es perdurable".

4.2.3.2 Lematizar

Para lematizar los documentos, se emplea un nuevo método en Python que funciona de manera parecida con la descripción del punto 4.2.3.2 con la diferencia de que el corpus de datos que se emplea para procesar cada palabra de cada documento es proporcionado por [62] y cuanta con más de medio millón de términos que ayudan en múltiples labores del procesamiento de datos.

A continuación, se muestra el texto después de Lematizar.

" vivienda ecológico zona semidesértica coahuila facultad arquitectura unidad saltillo universidad autónomo coahuila desarrollar proyectar vivienda ecológico zona semidesértico ejido narigua municipio general cepeda coahuila mediante usar arquitectura tierra apoyar reforzar tecnología sustentable zona semidesertica 12 importante ser respetar mediar ambientar alterar vivienda amigable confort persona ejido proyectar investigar estar proponer reducir cantidad material energía utilizar extraer recurso natural estar dejar lado arquitectura principio arquitectura tierra estar comprobar ser perdurar".

En este proyecto no se tomaron en cuenta conceptos de múltiples palabras como Universidad-Autónoma-Coahuila, ya que la jerarquía Wordnet no contiene ese tipo de término y, por ejemplo, la palabra universidad perdería presencia en la clasificación; si se deja como palabra unitaria puede ser comparada con otras palabras.

4.2.3.3 Aplicación de Sinónimos

Para el análisis de texto, resulta útil homologar ciertos términos con en el fin de darle más poder a la cuantificación de cierto tema. No es lo mismo encontrar en un documento las palabras "tele", "televisión", "tv" que tener tres veces la palabra "televisión", cuando se habla de análisis máquina.

Esta labor se consigue mediante otro método. Se lee cada palabra de cada documento y si cambia por su sinónimo si es que existe, de lo contrario se deja la palabra como se recibe. El diccionario de sinónimos se puede obtener de [73], aunque

no está completo; se añadieron nuevos términos manualmente. A continuación, se encuentra el texto después de aplicar sinónimos.

“ vivienda ecológica zona semidesértica coahuila facultad arquitectura unidad saltillo universidad autónoma coahuila desarrolla proyecto vivienda ecológica zona semidesértica ejido narigua municipio general cepeda coahuila mediante uso arquitectura tierra apoyada reforzada tecnologías sustentables zona semidesertica 12 importante es respetar medio ambiente alterarlo vivienda amigable confort personas ejidos proyecto investigar estamos proponiendo reducir cantidades materiales energía utilizada extracción recursos naturales estamos dejando lado arquitectura principios arquitectura tierra está comprobada es perdurable”.

4.2.4 Implementación del modelo LDA

Toca el turno a la agrupación de los documentos, para ello se empleará el algoritmo LDA. Para ejecutar este algoritmo se recurre a un módulo llamado LDA y puede encontrarse en [74].

El módulo LDA de Python requiere de tres variables y tres archivos creados a partir de los documentos con los que se alimenta el algoritmo.

Las tres variables son:

1. Número de tópicos (X): Es el número de cúmulos que se le pide encontrar el LDA.
2. Número de iteraciones (i): El número de veces que se repetirá la búsqueda apropiada para un cúmulo por documento. Mientras más grande el i, mayor el tiempo de ejecución.
3. Número de palabras que representan a cúmulo de palabras (P): Es la longitud del conjunto de palabras que el LDA emplea para representar a cada uno de los k cúmulos encontrados.

Los tres Archivos generados por conjunto de datos son:

a. Titles

Contiene todas las noticias ordenadas en el mismo archivo; se pone el número cero al principio del archivo y van aumentando en una unidad conforme se cambia de renglón. A un lado del número separado por un espacio el cuerpo completo de la noticia. Un salto de línea es la separación entre un documento y otro (Véase

Figura 4.2). El archivo ".titles" se crea a partir de un método en Python. A continuación, un ejemplo de ".titles" (las noticias aparecen con el tratamiento descrito en la sección anterior):

```
23 héctor santiago acordar angelino 1 año $5 millón
24 licey superar león suma 9 triunfo fila playoffs
25 césped gallardo davis seguir desempleado faltar
26 terdoslavich llevar licey 11 triunfo hilar seria
```

Figura 4.2: Fragmento de un archivo .'titles' obtenido a partir de un conjunto de noticias.

b. Tokens

El archivo '.tokens' es un diccionario de todas las palabras de todos los documentos con que se alimenta al modelo LDA. Las palabras son ordenadas alfabéticamente Véase Figura 4.3). El archivo ".tokens" se crea a partir de un método en Python.

```
7083 prometedor
7084 promocional
7085 promoción
7086 promover
7087 pronto
7088 pronunciar
7089 pronunciarse
7090 pronóstico
7091 propagación
7092 propagar
7093 propenso
```

Figura 4.3: Fragmento de un archivo '.tokens' obtenido a través de un conjunto de noticias.*Los números del lado izquierdo pertenecen al programa con el que se visualizaron, es decir, no pertenecen al archivo '.tokens'.

c. ldac

El archivo con extensión '.ldac' contiene el número de palabras en el documento; además, se agrega la posición de las palabras ordenadas alfabéticamente en el archivo '.tokens' junto en número de repeticiones de la palabra en el documento separado por dos puntos. Cada línea representa un documento (Véase Figura 4.4). El archivo ".ldac" se crea a partir de un método en Python.

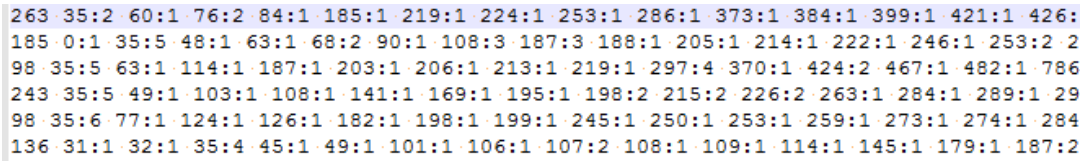


Figura 4.4: Fragmento de un archivo ".ldac" obtenido a partir de un conjunto de noticias.

La primera salida directa del algoritmo son los cúmulos de palabras, en la lista de abajo se delimitó el número, ya que para este ejercicio se usaron 50 palabras. A continuación, un ejemplo:

Palabras de cada tópico 0: ser basura suecia residuo país energía reciclaje noruego…
Palabras de cada tópico 1: ser sirio haber país estar entrar ir unir refugiado contra…
Palabras de cada tópico 2: museo ser louvre haber arte francés millón obra entrar…
Palabras de cada tópico 3: año ser ir 1 liga tener temporada carrera béisbol seriar…
Palabras de cada tópico 4: ser poder planeta haber estar galaxia estrella tierra hacer…

La segunda salida es la clasificación de documentos, por ejemplo:

Número tópico documento 4: 353 galaxia luminoso universo autodestruye
Número tópico documento 0: 287 suecia necesitar residuo suecia necesitar
Número tópico documento 1: 232 isis cambiar parámetro trabajar

Nota: La primera línea muestra que se asignó al cúmulo 4 el documento 353. Las palabras que continúan después del número de documento (en este caso 353) son las palabras que representan a cada cúmulo.

4.2.5 Traducción para acceder a Wordnet

Como se ya ha mencionado, en este trabajo se clasifican documentos en español, por lo tanto, se requiere convertir los términos de las bolsas de palabras provenientes del algoritmo LDA (etapa anterior) al lenguaje inglés, y con ello obtener un representante en la jerarquía de Wordnet. Empero, traducir palabras resulta una tarea complicada debido a la ambigüedad, prácticamente se convierte a la desambiguación en una herramienta imprescindible para laborar con Wordnet. En este trabajo más que trabajar con oraciones bien estructuradas como: "puse la carta en el sobre y la entregué al cartero", se trabaja con bolsas de palabras cuyo orden de aparición está en función de su probabilidad de ocurrencia, y no una oración bien estructurada, como ejemplo la siguiente bolsa de palabras: "sobre ser enviar carta haber estar paquete calle tierra negro hacer año científico cuartilla lápiz". Claramente puede observarse que las palabras de la bolsa no tienen una estructura

bien definida. Afortunadamente, si se tiene contexto que puede explotarse para desambiguar. A continuación, se presenta un algoritmo de desambiguación útil al trabajar con bolsas de palabras (Véase Algoritmo 1).

Algoritmo 1. Traducción con desambiguación de bolsas de palabras

Entradas: palabra a desambiguar, bolsa de palabras, diccionario para traducir.
Salida: Palabra traducida desambiguada

1. Contador[S]←0
2. Para cada palabra en la bolsa de palabras
   a. Para cada solución en el diccionario para traducir
      i. Si la palabra está en el contexto de la solución S'
         i.ii. Contador[S'] += 1
3. Regresar la solución con el puntaje más alto

El diccionario para traducir tiene la siguiente estructura (Véase Figura 4.5):

WTD: Solución$_1$ {palabra$_{11}$, palabra$_{12}$, …}
WTD: Solución$_2$ {palabra$_{21}$, palabra$_{22}$, …}
WTD: Solución$_3$ {palabra$_{31}$, palabra$_{32}$, …}
WTD: Solución$_h$ {palabra$_{h1}$, palabra$_{h2}$, …}

Figura 4.5: Estructura del diccionario de traducción.

WDT es la palabra a traducir y desambiguar; después de los dos puntos están las soluciones para WDT y entre llaves se encuentran el contexto. El contexto en este caso, es un conjunto de palabras clave que ayudan a distinguir la solución actual de las demás soluciones; el contexto puede estar formado una o múltiples palabras con una única condición: $cont_w \cap cont_z = \emptyset$, donde, $w, z = 1,2, \dots, h$ y $w \neq z$. Esta condición indica que la desambiguación se cumplirá siempre y cuando las palabras clave (contexto) que representen a las soluciones no tengan palabras en común.

Más adelante se hará alusión a esta etapa (Sección 4.2.6), sin embargo, es importante mencionarla en este punto para facilitar la comprensión.

4.2.6 Cálculo de los cúmulos naturales

Para encontrar el número de cúmulos naturales (K) se analizan las bolsas de palabras que se obtienen como salida del algoritmo LDA. Lo que se hace es solicitarle al LDA que agrupe los documentos en X= *2, 3, 4, …, lim* grupos y con ello analizar las bolsas de palabras que representan a cada grupo X, posteriormente se mede la distancia semántica entre los términos de las agrupaciones para calcular la compacidad y lejanía entre cúmulos, con estos valores es posible encontrar el K deseado.

4.2.6.1 Cálculo de compacidad (distancia promedio intra-cúmulo)

Se emplea Wordnet para calcular la similitud de todas las palabras que representan a cada uno de los cúmulos que se obtiene del algoritmo LDA. La compacidad, o la distancia promedio intra cúmulo, indica que tan compacto es un cúmulo. Mientras más pequeños sean los valores de las intra-distancias puede decirse que es un cúmulo bueno (“ya que la relación semántica entre ellos es mayor”). Aunque, cabe aclarar que pueden existir configuraciones de cúmulos naturales en los conjuntos de dato, sin embargo, para elegir un mejor k es fundamental usar también las inter-distancias (Sección 4.2.6.2).

La compacidad de un grupo se define como:

$$Com(A) = \frac{\sum_{a,b \in A,\ a \neq b} dist(a,b)}{(n*n-1)/2}$$ Definición1

Donde *dist* es *1-sim(),* a,b son palabras pertenecientes al conjunto A, n es la cantidad de palabras en el conjunto A.

Para calcular la compacidad de los grupos de palabras que se reciben del algoritmo LDA se emplea la Definición 1. Las palabras se encuentran en español por lo que debe hacerse la traducción de cada término al idioma inglés (el idioma con el cuál puede accederse a la jerarquía hiperónimo/hipónimo de Wordnet). Para realizar la traducción se empleó un diccionario web español-inglés [75] que contiene

miles de términos en español con sus correspondientes traducciones. Al diccionario fueron agregados nuevos términos manualmente para ampliar su poder. Para desambiguar las palabras y hacer más consistente la traducción se empleó el Algoritmo 1 (Véase sección 4.2.5). Una vez que se tradujo cada palabra se cambió por su correspondiente Synset [34]. Si una palabra ordinaria tiene más de un Synset que la represente, se selecciona el primero de ellos para calcular la distancia semántica. Esto funciona ya que todos los Synsets de esa palabra están semánticamente relacionados; la distancia entre ellos es mínima ya que son hermanos en la taxonomía de Wordnet, por lo tanto, a pesar de que puede haber diferencias entre los valores de similitud, estas diferencias no afectan el cálculo de la compacidad o lejanía entre grupos. Por ejemplo, la distancia del primer Synset de *Tiger* con todos los Synsets de *Horse* tienen como valores de similitud {0.08333,0.09090, 0.083333,.1 ,0.08333,0.09090}; cómo puede verse son valores muy parecidos. Una vez que se obtiene los "Synsets" para cada palabra se procede a calcular la similitud entre palabras de cada cúmulo. Eso se hace calculando la similitud de cada palabra del mismo cúmulo. Para esta tesis se empleó la función de similitud "Path" que proviene de [35] y [36]. La función Path regresa un valor denotando que tan similares son los sentidos de un par de palabras basado en el camino más corto que conecta los sentidos de las palabras en la taxonomía hiperónimo/ hipónimo [76]. En el capítulo 5 (Pruebas se menciona porque se empleó este tipo de medida de similitud entre palabras). Véase sección (3.11.1) para más detalles de la similitud Path. Una vez calculada la compacidad de cada partición se saca el promedio; este promedio obtenido compacidad promedio.

Las similitudes de las palabras que pertenecen a un mismo cúmulo tienen valores entre cero y uno.

La Figura 4.5, muestra cuántas medidas de similitud se calculan para el cúmulo c_3 (aunque lo mismo debe hacerse para cada conjunto), claramente, puede observarse que la cantidad de medidas calculadas para cada cúmulo es de:

$$\binom{n}{2} = \frac{n*(n-1)}{2} \qquad \text{Fórmula 4.2.1}$$

La cantidad de similitudes calculadas a primera vista parecieran ser un problema debido a la cantidad de palabras en los documentos, no obstante, no lo son, ya

que la cantidad de palabras que representan a cada cúmulo siempre está bajo el control del usuario (no suele ser muy grande), como ya se ha mencionado al inicio del capítulo.

En la Figura 4.6 cada flecha representa el cálculo de similitud entre pares de palabras del mismo cúmulo, se hace lo mismo para cada cúmulo.

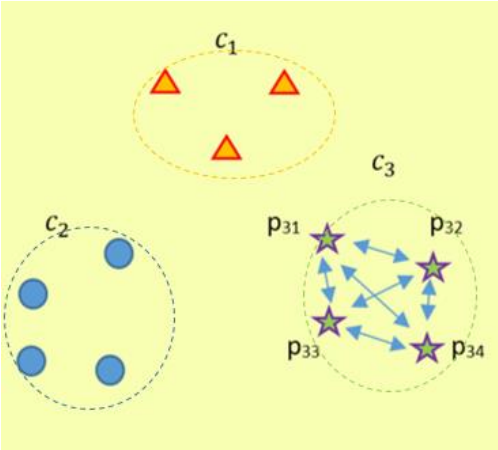


Figura 4.6: Número de medidas de similitud calculadas para el cúmulo c_3.

4.2.6.2 Cálculo de la lejanía entre cúmulos (distancia promedio inter-cúmulo)

Se emplea Wordnet para calcular la lejanía entre cúmulos. Entre más grande sea la distancia que existe entre un par de cúmulos, más diferente será su contenido y viceversa.

La lejanía entre grupos se define como:

$$lej(A,B) = \frac{\sum_{a\in A, b\in B} dist(a,b)}{nm}$$ Definición 2

Donde n y m son las palabras de los cúmulos A y B respectivamente.

Para realizar el cálculo de las lejanías naturales entre cúmulos lo primero que debe realizarse es la elección de dos de ellos. Una vez más, la cantidad de lejanías naturales estará enmarcada por una combinatoria $\binom{X}{2}$, donde X es el número de cúmulos.

Los conjuntos de palabras sufren la misma modificación que las palabras descritas en la sección anterior (4.2.6.1). Esto es, traducción al idioma inglés y obtención de los Synsets para cada palabra en los conjuntos. La distancia entre dos palabras surge de la idea, “si dos palabras son similares en un valor s (“s” es el valor que se obtiene de una función de similitud), entonces la distancia entre dos palabras es uno menos la similitud”, es decir:

dist=1-similitud Fórmula 4.2.2

La función de similitud “Path” nuevamente fue la que se ocupó para esta labor.

En la Figura 4.7, puede verse la cantidad cálculos de lejanías entre palabras necesarias para obtener la lejanía natural entre los cúmulos c_3 y c_4.

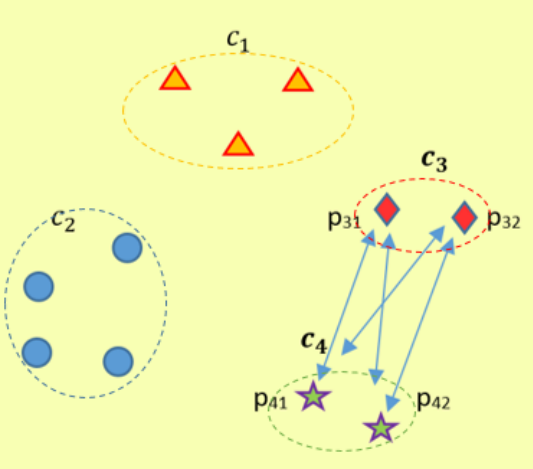


Figura 4.7: Representación del número de similitudes que deben calcularse entre un par de cúmulos para Saber la lejanía natural entre los cúmulos c_3 y c_4.

4.2.6.3 Determinando las agrupaciones naturales K

Una vez calculados los promedios de las compacidades y de las lejanías naturales para cada X, se define a K como:

$$k = \arg\_min_{x\epsilon[1,lim]} C_x(P)/\ell_x(P)$$ Definición 3

El K buscado es el *Argumento Mínimo* de los valores del *Cociente* entre el promedio de las compacidades y las lejanías naturales. La fórmula anterior surge de querer obtener un valor pequeño que ayude a saber el K natural y sabiendo que la compacidad es pequeña y la lejanía es grande la forma de relacionarlos es que mediante una división o cociente; se tienen *X* cocientes por lo que se elige el mínimo, no obstante, lo que interesa de ese valor mínimo es saber su posición o el argumento mínimo (Véase Figura 4.7).

Para ejemplificar el procedimiento se tiene la Tabla 4.2 que muestra los valores de compacidades y lejanías (sólo con fines de ejemplificación) para diferentes particiones de cúmulos X_i, donde X_i= {2, 3, 4, 5, 6, 7}. Como se mencionó anteriormente el promedio de las intra-distancias es un valor pequeño y el promedio de las inter-distancias es un número grande.

Tabla 4.2: Ejemplo de promedios de intra e inter distancias para X=2, 3, 4, …, lim

Promedios Intra-distancias	Número de cúmulos (x=)	Promedios Inter-distancias
1.8	2	200.13
4.3	3	180.58
2.7	4	358.15
0.9	5	400.13
5.3	6	317.98
6.2	7	350.12

Entonces, substituyendo los valores de la Tabla 4.2 en la Definición 3 se tiene:

$$K = \left[\arg\_min\left(\frac{1.8}{200.13}, \frac{4.3}{180.58}, \frac{2.7}{358.15}, \frac{.9}{400.13}, \frac{5.3}{317.98}, \frac{6.2}{350.12}\right)\right]$$

$$K = [\arg\_min(.00899, .02381, .00753, .00224, .01666, .01770)]$$

Ya que se comenzó de un x=2, la primera posición corresponde al x=2, la segunda x=3, etc. Por lo tanto, para los valores de la Tabla 4.2 el K natural es: K=5
En la sección 5.2 hay múltiples pruebas que se hicieron con documentos reales. En esos ejercicios se comprueba que el modelo funciona adecuadamente, es decir, se encuentra el número de cúmulos naturales con mayor relación semántica. El K que indica los cúmulos naturales puede no ser único, es decir, puede haber más de un K que tengan sentido para los mismos documentos (Véase sección 5.5).

4.2.7 Etiquetación de cúmulos

Ya que se han encontrado los cúmulos naturales, se procede a etiquetarlos. Una buena etiquetación debe ser fácilmente comprensible, informativa y representativa [9]. Partiendo de este comentario se construye un método de etiquetación usando Wordnet.

En seguida se muestra el algoritmo de etiquetación (Algoritmo 4):

Algoritmo 2. Etiquetado jerárquico usando múltiples palabras

Entrada: Conjunto de k bolsas de palabras conteniendo n palabras
Salida: Las U palabras más relevantes

1. Para cada bolsa de palabras en bolsas de palabras
    a. Para cada palabra de bolsa de palabras
        i. Traducir palabra a inglés
        ii. Obtener el Synset representativo de la palabra
        iii. Medir la profundidad del Synset en la taxonomía semántica
    b. Ordenar los Synsets de acuerdo con su valor en orden descendente.
    c. Synsets-relevantes ← Los U Synsets más relevantes
    d. Para cada Synset en Synsets-relevantes
        i. Obtener la palabra natural del Synset
        ii. Traducir palabra a español

Explicación del algoritmo de etiquetación:

El Algoritmo 2, toma como entrada k bolsas de palabras salientes del algoritmo LDA y un número U de palabras que se deseen encontrar; para las pruebas realizadas en la sección IV se utilizaron etiquetas con $U = 3$. El hecho de elegir un número de palabras mayor a la unidad para representar a los conjuntos es que una única palabra es muy general para representar el contenido de todo un conjunto de datos. La traducción de 1.a.i, se hace empleando el Algoritmo 1. En 1.a.iii se obtienen las profundidades de cada Synset del cúmulo; lo que se busca con la profundidad y el ordenamiento de la instrucción 1.b es que los U términos con mayor peso en la jerarquía elegidos en 1.c sean los que representen a cada cúmulo. Finalmente, en d.i y d.ii se recuperan las palabras del cúmulo que sufrieron el cambio a Synsets para que sean nuevamente palabras en español.

4.2.8 Visualización de polígono

El proceso de visualización es la etapa que muestra gráficamente los resultados de las diferentes etapas del sistema; en los siguientes puntos se explicará la forma en que se trataron los datos para poder realizar las visualizaciones. Cabe aclarar que las visualizaciones en este proyecto se programaron individualmente, es decir, no se empleó ningún tipo de software especializado, todo se hizo artesanalmente. Incluso existen visualizaciones novedosas que se crearon especialmente para apoyar al reconocimiento de tópicos naturales. Se empleó un módulo en Python llamado *Turtle* que permite graficar con base en las posiciones de la pantalla.

El polígono de la Figura 4.8 es una representación visual compleja que muestra algunas propiedades de los cúmulos; en ella se reflejan múltiples parámetros que a continuación se describen.

4.2.8.1 Los círculos

Cada círculo representa un conjunto de palabras, es decir, un cúmulo. El tamaño de los círculos es proporcional al número de documentos asignados a ese conjunto, es decir, el conjunto con mayor número de documentos asignados será el más

grande y el que tenga menor número será el más pequeño. Sin embargo, los círculos tienen una modificación mediante código, ya que es de esperarse que sí se tiene una cantidad de documentos muy grande los límites de los círculos podrían salirse de la pantalla y deformar la imagen. Tomando en cuenta que una pantalla moderna de última generación sólo puede llegar a tener por mucho 4000 pixeles, el proyecto se vería forzado a trabajar con un máximo de 4000 documentos sí no se usara normalización; es por ello que se debe asistir a un modelo matemático para solventar la situación.

La normalización del radio ayuda a que la visualización tenga un comportamiento controlado y así evitar problemas; la visualización depende de la pantalla para que puede ser representada adecuadamente, y debe considerarse el tamaño de la misma, pues pueden presentarse casos en los que se tengan miles de documentos y sí no se normaliza el despliegue de la información se saldrá de la zona de representación ya el radio de los círculos que representan cada tópico está en función de la cantidad de documentos clasificados para ese tema; tomando en cuenta que una pantalla moderna de última generación sólo puede llegar a tener por mucho 4000 pixeles, el proyecto se vería forzado a trabajar con un máximo de 4000 documentos sí no se usara normalización, es por ello que se debe asistir a una representación matemática para solventar la situación.

4.2.8.2 Procedimiento de Normalización

Se tienen un conjunto de valores de la forma, donde los x_i son los valores de los radios de los círculos actuales, K es el número de cúmulos que se tienen:

$$valores = \{x_1, x_2, x_3, \dots, x_k\} \qquad \text{Fórmula 4.2.5}$$

También se tienen dos nuevos valores que determinan el nuevo mínimo N_{min}y el nuevo máximo N_{max} del radio deseado

Se deben calcular dos rangos, uno para el conjunto de valores y otros para los nuevos valores, esto es,

$$R_o = MAX(valores) - MIN(valores) \qquad \text{Fórmula 4.2.6}$$

Y

$$R_n = N_{max} - N_{min}$$ Fórmula 4.2.7

También se debe obtener un factor de multiplicación para la nueva escala empleando los rangos obtenidos:

$$FM = \frac{R_n}{R_o}$$ Fórmula 4.2.8

Para normalizar el radio debe usarse la siguiente fórmula

$$x_i' = N_{min} + \big(x_i - MIN(valores)\big) * FM$$ Fórmula 4.2.9

Finalmente, los nuevos valores quedan como:

$$valores' = \{x_1', x_2', x_3', \dots, x_k'\}$$

Lo que se busca con este procedimiento es que el cúmulo con menor número de documentos asignado tenga un radio mínimo controlado; además, que el tamaño del cúmulo más grande sea el valor máximo de los radios y que los radios de los círculos que están entre los valores máximos sean proporcionales a su tamaño.

4.2.8.3 Relaciones continuas y no continuas (líneas entre cúmulos)

Las relaciones continuas son aquellas líneas que unen el centro de los círculos contiguos. Cada línea que representa una relación tiene un número que muestra la cantidad de elementos que comparten, se calcula mediante los elementos que se intersectan entre los conjuntos de palabras. Para el caso de las relaciones no contiguas se presentan los mismos criterios de visualización que para las continuas con la única diferencia de que las líneas se desplazan hacia cúmulos no contiguos.

Las representaciones de las relaciones entre cúmulos también pueden ocultar el número junto a cada una de las líneas ya que pueden ser poco intuitivos. Luego entonces, la visualización puede cambiar el grosor de las líneas para que representen este tipo de relaciones en forma visual con líneas y no números (Véase Figura: 4.8). Así, mientras más grande sea el número de palabras que comparten dos cúmulos, mayor será el grosor de las líneas y viceversa.

4.2.8.4 Indicadores de comienzo

El primer indicador de comienzo surge con el círculo inicial el cuál se ubica del lado derecho de la pantalla, sí se tuviera un eje imaginario de ordenadas sería el primer círculo al que se pudiera llegar conforme crece el número de grados en contra de del sentido de las manecillas del reloj. Sirve para saber cuál es el primer tópico encontrado en el conjunto de datos.

El segundo indicador, pertenece a cada cúmulo; este consiste en una punta de flecha Indicando la palabra con mayor relevancia en el cúmulo; después de esto, las palabras se despliegan en sentido de las manecillas del reloj en orden descendiente de relevancia. Aquí, se define “relevancia” como la popularidad de la palabra en los documentos que forman el cúmulo. Mientras más veces aparezca la palabra (o sus sinónimos y lemas), tanta más relevancia tiene. Esto es relevancia absoluta, y tiene el problema de que depende del número total de palabras en el cúmulo. La relevancia relativa es el porcentaje o fracción de aparición de esa palabra entre todas las palabras del cúmulo, es la relevancia absoluta dividida entre el número de palabras del documento, es un número entre 0 y 1.

4.2.8.5 Palabras en la representación visual

- Palabras en los círculos

Para visualizar las palabras en cada cúmulo se le pide al algoritmo LDA cierta cantidad de términos, por ejemplo, 50. Sin embargo, en cada cúmulo se puede notar que el número de palabras desplegadas es muy pequeño. Esto obedece a que se realizan varias actividades para elegir palabras.

Cómo primera acción se puede identificar que el tópico con mayor presencia (Definida en 4.2.8.5) entre los documentos, es aquel que contiene mayor número de palabras, para la Figura 4.8 se eligieron 10 palabras. Después, se puede identificar que conforme disminuye la presencia de cada tópico en los documentos también disminuyen las palabras desplegadas en cada cúmulo (círculo).

La elección de las palabras está en función de la probabilidad con la que el algoritmo las ordena. Además, se envían al final de las listas de palabras los verbos al

final de cada bolsa de palabras. Esto se hace ya que en algunas ocasiones las palabras que encabezan las listas de los cúmulos son verbos y si se dejaran como las entrega el algoritmo LDA dificultarían el proceso de visualización ya que las palabras predominantes serían los verbos (cabe aclarar que las palabras no se modifican ni eliminan, "sólo se llevan al final en la visualización"). A continuación, se presenta el Algoritmo 3 para cambiar el orden de aparición de los verbos de las listas de palabras en la visualización.

Algoritmo 3: Priorización de palabras en la visualización.

Entradas: $list_{verbos}$, obtener un $list_{bolpal}$.

Para cada $bolsa$ en $list_{bolpal}$

    Para cada $palabra$ en $bolsa$

        Si $palabra$ contenida en $list_{verbos}$

            Enviar palabra al final de la $bolsa$

$list_{verbos}$: Es una lista de verbos simple, que contiene una raíz de los verbos en español.

$list_{verbos}$: Es el conjunto de bolsas de palabras con el conjunto de palabras que regresa el algoritmo LDA ordenada de acuerdo a la probabilidad de ocurrencia

Si se desea priorizar un conjunto de palabras como las *nacionalidades,* para que se pongan en primer lugar, lo que debe realizarse es incluir una condición similar al Algoritmo 1 pero guardando las palabras que cumplan con la condición al principio.

- Listas de palabras completas

El ver pocas palabras que representan a los cúmulos en los círculos como se explicó arriba, ayuda a que un usuario comprenda más rápido sobre el contexto general del cúmulo. No obstante, es conveniente poder mostrar una lista más amplia de palabras como puede verse en la Figura 4.9; la lista de las palabras que aparecen en la figura puede observarse al seleccionar su etiqueta, la lista aparecerá en

la parte derecha de la visualización con todas las palabras que representan al cúmulo.

4.2.8.6 Colores

Los colores usados para rellenar los círculos varían de acuerdo con su temperatura, es decir, en colores cálidos y fríos. Los colores cálidos se ocupan para identificar a los tópicos con más presencia de documentos y los fríos para los de menor presencia. Las relaciones continuas y no continuas poseen colores tenues para no confundirse con las letras en color negro.

4.2.8.7 Posicionamiento de las etiquetas

Las etiquetas se colocan del lado izquierdo de la visualización. Muestra el número de tópico al que representan iniciando desde cero. Se eligió ese lado de la pantalla para no interferir con el crecimiento de la lista completa de palabras pertenecientes al cúmulo.

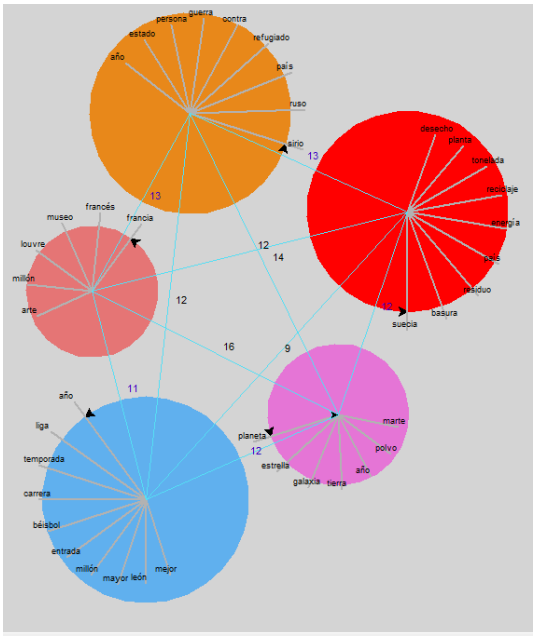


Figura 4.8: Visualización de polígono. Cada círculo representa un cúmulo. Los conjuntos de noticias representados son: Suecia, Siria, Louvre, Béisbol y Universo.

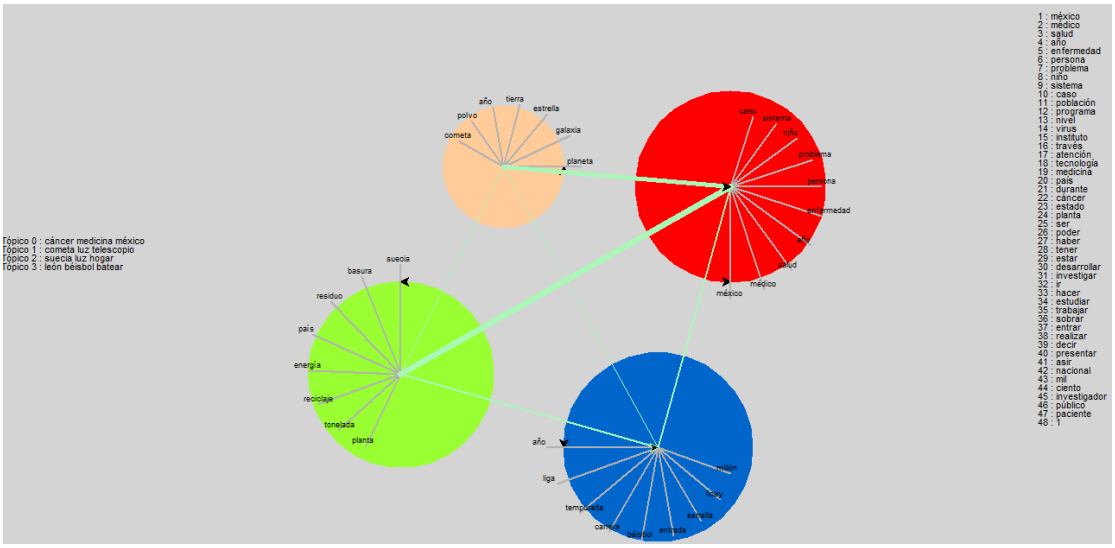


Figura 4.9: En la imagen se observa en la parte central cuatro cúmulos representados por círculos, entre más grande el círculo mayor es la cantidad de documentos que lo conforman; las líneas entre los círculos representan las palabras que comparten los dos círculos que se unen por ella. Las palabras a la izquierda son las etiquetas para cada cúmulo, la primera etiqueta corresponde al primer círculo empezando en el círculo con una punta de flecha en su interior, posteriormente se avanza en contra de las manecillas del reloj. La lista de palabras a la derecha son las palabras que representan un cúmulo, surgen al seleccionar una etiqueta.

5. Pruebas y resultados

En esta instancia se presentan las pruebas hechas al modelo creado.

Todos los ejemplos se realizan con cincuenta palabras, en la sección 5.5 puede verse la razón de este número.

5.1 Pruebas del modelo LDA usando el tratamiento de lenguaje natural

La siguiente prueba se realizó con los conjuntos de datos: Béisbol, Louvre, Salud, Suecia y Universo.

A continuación, se presenta la salida del algoritmo LDA sin ninguna de las etapas del tratamiento de lenguaje natural. Como puede observarse las stop-words son las que predominan en cada cúmulo, nótese que palabras como ‘En’ y ‘en’ a pesar de que para los seres humanos es lo mismo (sólo por estar en mayúsculas), para una computadora son palabras completamente diferentes.

Palabras de cada tópico 0: de la que en el y a los del se las para con por un es una no como al su o lo más este son salud personas México Nacional mil Salud esta año años casos ciento población sobre está enfermedad han virus donde niños puede sus tiene investigación
Palabras de cada tópico 1: de en el y la los que con a un del para una se por su al las como dos fue más años temporada lo Liga no béisbol es primera año tres El sus primer desde equipo Licey dijo San quien Nacional Grandes Estrellas cuatro carrera millones uno Ligas
Palabras de cada tópico 2: de la que en el y a un del los se una por es las más con para su como lo al El no ha este planeta sus Tierra La tiempo años son negro polvo desde muy ser puede estrellas si agujero pero Los Marte luz hasta galaxia En
Palabras de cada tópico 3: de la el que en y del a los un una las por Louvre su se con El al más museo para La ha no es como Museo sus Francia entre millones o arte visitantes cultura En exposición sobre obras lo mundo París hoy obra francés Lisa dos pintura
Palabras de cada tópico 4: de que la en y los el se a para basura es un Suecia residuos por una las energía con más reciclaje del países su al no sus como Noruega país toneladas o ha lo incineración desechos plantas suecos todo otros calefacción ahora hogares electricidad le La tienen Y

Ahora se presentan los mismos cúmulos que se presentaron arriba eliminando stop-words y haciéndolas minúsculas. Puede verse claramente que la salida del LDA puede entenderse claramente.

Palabras de cada tópico 0: es tierra planeta sus negro polvo son agujero marte tiempo luz estrellas universo años galaxia está puede solar nasa hay galaxias gran mil estrella han millones entre tiene cometa podría venus alrededor planetas superficie hace datos astrónomos veces órbita científicos fue posible espacio kilómetros sobre podrían año primera forma

Palabras de cada tópico 1: basura es suecia residuos energía reciclaje países sus noruega país toneladas incineración desechos suecos plantas calefacción hogares electricidad tienen producir son importar capacidad sistema reciclar plástico resto proceso agencia mil oslo han esto recicla eso habitantes casi año están desperdicios tiene sido importa metales ciudad casas vertederos medio además
Palabras de cada tópico 2: museo louvre es arte entre francia cultura millones sobre visitantes exposición hoy mundo obras francés parís obra han lisa año pintura siglo durante muestra cultural mil público era director dos años nueva retrato fue tres tras mona pasado da leonardo parte museos hace gran artistas francesa solo historia
Palabras de cada tópico 3: dos fue años béisbol temporada es liga año serie grandes ligas primera tres san carrera nacional equipo estrellas primer licey dijo uno millones cuatro carreras segundo jugador contrato tigres bateó mundial americana leones acuerdo premio toros 11 carlos nueva tuvo sobre tiene mayores jugadores céspedes york votos entrada
Palabras de cada tópico 4: es salud son nacional méxico personas mil investigación año casos años sistema ciento población hay virus puede está sus sobre enfermedad niños programa atención han nivel través tiene así enfermedades durante medicina dos desarrollo tecnología estudio instituto problemas obesidad cáncer evitar además pueden entre parte diagnóstico acuerdo dengue doctor

Finalmente, se presentan los cúmulos de palabras con el tratamiento de lenguaje natural usado en este proyecto, es decir, remover stop-words, aplicar sinónimos y lematizar.

Palabras de cada tópico 0: ser poder salud haber tener médico año estar nacional persona enfermedad méxico investigar desarrollar mil ir hacer problema trabajar niño estudiar sistema ciento caso sobrar población investigador programa nivel virus realizar país atención instituto través 1 decir entrar asir medicina tecnología durante mayor estado cáncer presentar mejorar parte
Palabras de cada tópico 1: museo ser louvre haber entrar arte obra cultura francés millón tener año francia sobrar visitante exposición hoy planeta dar estar ir siglo hacer venus mostrar poder público mil ver unir artista parir decir pintura explicar lisa durante cultural tres retratar recibir 1 director crear grande nueva abrir
Palabras de cada tópico 2: ser basura suecia residuo país energía reciclaje noruego tonelada producir su tener sueco incinerar planta importar reciclar generar desecho estar hogar poder año separar calefacción plástico procesar hacer ciudad luz convertir lograr ir metal haber 1 unir vertedero bolsa restar capacidad mil color sistema este llevar agencia deber recibir
Palabras de cada tópico 3: año ser liga 1 tener ir temporada béisbol grande estar jugador haber nacional unir seriar su carrera decir millón equipar mayor poder rojo acordar jugar san batear contratar premiar media incluir mejor 2 anunciar americana tres último ganar mundial llegar próximo juego york nueva 3 salón hacer césped
Palabras de cada tópico 4: ser poder estrella planeta haber estar galaxia tierra ir negro año solar hacer 1 polvo su marte cometa agujerar tener luz universo tiempo encontrar telescopio observar científico mil nasa estudiar licey astrónomo órbita decir llegar carrera entrar unir explicar kilómetro 2 gran gas alrededor forma cuatro conocer investigador

Con esta prueba se concluye que el tratamiento del lenguaje natural es sumamente útil para encontrar los lemas y con ellos, los cúmulos naturales en documentos; además, permite encontrar de manera clara las palabras representativas de los cúmulos y ayuda a tener homogeneidad de términos.

5.2 Pruebas de la etiquetación de los cúmulos

Ahora se exponen las pruebas de las etiquetas que distinguen a los cúmulos en cada configuración. En un principio de muestran todas las palabras que representan a cada cúmulo, después, solamente se muestran las palabras de le etiqueta y el número de conjuntos para los que se obtuvieron.

a) Las etiquetas que se obtuvieron con los conjuntos de datos Corea, Louvre, Salud, Siria, Suecia, Universo son:

Para 4 cúmulos, todas las palabras de los conjuntos

Palabras de cada tópico 0: ser basura suecia residuo país energía reciclaje noruego tonelada producir sueco su incinerar tener planta importar reciclar generar estar desecho hogar poder hacer calefacción separar año plástico luz convertir ir procesar ciudad lograr haber metal unir 1 bolsa vertedero capacidad color papel mil este restar sistema recibir llevar agencia
Palabras de cada tópico 1: ser poder haber estar tener año salud planeta mil hacer médico ir 1 estudiar estrella nacional galaxia tierra su sistema investigar investigador enfermedad sobrar persona decir méxico trabajar negro científico entrar solar desarrollar tiempo encontrar mayor conocer deber problema millón instituto usar polvo parte caso durante observar realizar
Palabras de cada tópico 2: ser sirio haber su país entrar ir estar contra unir refugiado guerra año persona estado militar tener hacer poder sobrar llegar gobernar morir pasado atacar isis ciudad grupo parte ayuda ruso internacional decir civil mes organización tras tres vida humanitario solo 1 localidad niño terrorista encontrar ejército muerte
Palabras de cada tópico 3: ser museo haber corea louvre unir probar nuclear norte su entrar estado bomba poder país norcoreano año obra arte tener francés hacer planeta 1 francia dar ir pyongyang seguridad estar chino sobrar cultura millón hoy visitante nueva mostrar sur exposición hidrógeno decir internacional gran kim ver siglo pasado

Para 4 cúmulos, las etiquetas son:

Tópico 0: Suecia, luz, hogar
Tópico 1: México, encuesta, estigmatizar
Tópico 2: Siria, ciudad, militar
Tópico 3: Exposición, Francia, franceses

Para 5 cúmulos, todas las palabras de los conjuntos son:

Palabras de cada tópico 0: museo ser louvre su arte haber obra francés entrar millón cultura año francia tener visitante exposición sobrar público ir dar siglo poder hoy durante parir mujer hacer director mostrar pintura lisa artista ver grande retratar mil sala 2 decir cultural tres 1 día abrir explicar gran planeta imagen
Palabras de cada tópico 1: ser poder año haber salud tener estar médico mil estrella estudiar ir galaxia nacional persona sistema investigar investigador enfermedad hacer planeta méxico su trabajar 1 desarrollar problema ciento entrar conocer realizar caso solar millón polvo instituto usar parte programa cometa centrar científico tierra ciencia explicar sobrar encontrar decir presentar
Palabras de cada tópico 2: sirio ser haber país su refugiado entrar contra guerra persona año ir unir morir isis estar llegar hacer atacar estado militar gobernar ciudad grupo civil ayuda parte humanita-

rio pasado mes localidad tener terrorista ejército último encontrar europa muerte organización millón irak tres turquía poder población vida niño sobrar
Palabras de cada tópico 3: ser basura suecia residuo país energía reciclaje noruego tonelada sueco producir tener su incinerar planta importar reciclar generar estar desecho hogar poder separar calefacción luz hacer plástico procesar año convertir lograr ciudad ir 1 metal haber unir bolsa vertedero color capacidad mil recibir sistema papel llevar restar este agencia
Palabras de cada tópico 4: ser haber unir corea probar poder estar estado norte nuclear planeta país hacer su bomba tener ir 1 internacional sobrar decir tiempo norcoreano entrar seguridad pyongyang chino ver contra sur después hidrógeno llevar anunciar pasado nueva gran año kim negro volver tierra estadounidense considerar crear llegar marte régimen solo

Para 5 cúmulos, las etiquetas son:

Tópico 0: Exposición, Francia, franceses
Tópico 1: Cometa, México, encuesta
Tópico 2: Turquía, Irak, Suiza
Tópico 3: Suecia, luz, hogar
Tópico 4: Marte, probar, internacional

Para 6 cúmulos, todas las palabras de los conjuntos son:

Palabras de cada tópico 0: ser haber su tener poder hacer año refugiado sirio llegar país sobrar tiempo ver ir persona entrar estar día antes vida europeo millón pasar tres trabajar niño guerra último nuevo bueno deber importante dejar permitir morir europa propio saber además volver mes parecer era bien plan parte futuro
Palabras de cada tópico 1: ser salud médico poder año nacional haber enfermedad méxico estar persona investigar tener mil desarrollar sistema ir ciento estudiar caso problema población programa virus nivel investigador realizar instituto 1 niño través medicina trabajar atención sobrar paciente cáncer durante ciencia asir país hacer tecnología diagnóstico mayor obesidad especialista presentar entrar
Palabras de cada tópico 2: museo louvre ser obra francés arte entrar año cultura francia haber visitante su millón exposición planeta siglo público parir mostrar tener hoy pintura dar artista retratar lisa mujer mil director estar recibir explicar sobrar cultural abrir crear 2 mono periodista lugar sala 1 alemán grande tres nueva poder
Palabras de cada tópico 3: basura ser suecia residuo país energía reciclaje noruego tonelada sueco producir incinerar su tener planta importar generar reciclar desecho estar hogar separar calefacción plástico procesar luz año convertir poder lograr ir ciudad metal hacer unir 1 vertedero bolsa capacidad color sistema mil haber este restar llevar recibir agencia papel
Palabras de cada tópico 4: ser poder planeta haber estar estrella galaxia tierra negro hacer año científico solar tener 1 polvo ir cometa marte agujerar luz mil universo observar estudiar telescopio decir tiempo nasa órbita su astrónomo encontrar explicar investigador millón conocer entrar imagen kilómetro alrededor superficie gas gran unir vez llegar venus sol
Palabras de cada tópico 5: sirio haber unir corea estado país ser probar contra norte nuclear entrar bomba estar su ir internacional norcoreano militar seguridad pyongyang 1 pasado régimen isis chino sur atacar nueva hacer sobrar decir hidrógeno poder ayuda ruso nación parte kim humanitario rusia ciudad terrorista considerar anunciar ejército organización guerra

Para 6 cúmulos, las etiquetas son:

Tópico 0: Siria, refugiado, estado
Tópico 1: Cáncer, medicina, México
Tópico 2: Mono, exposición, Francia
Tópico 3: Suecia, luz, hogar
Tópico 4: cometa, luz, telescopio
Tópico 5: Siria, ciudad, probar

Para 7 cúmulos las etiquetas son:

Tópico 0: Exposición, Francia, franceses
Tópico 1: Suecia, luz, hogar
Tópico 2: México, médico, enfermedad
Tópico 3: Luz, telescopio, agujerar
Tópico 4: Siria, Ciudad, poblar
Tópico 5: Cometa, marte, venus
Tópico 6: Medicina, México, estigmatizar

Nótese que las etiquetas mostradas en cada configuración de tópicos despliegan adecuadamente la representación de temas que representan. Usar una etiqueta de múltiples palabras permite identificar más fácilmente el tópico con mayor presencia en el cúmulo. Nótese que en estas pruebas se eligieron tres palabras, pero no se limita a esa cantidad; el número máximo de palabras de la etiqueta es el total de palabras en el cúmulo.
En este ejemplo, el número natural de cúmulos es seis; puede verse que la etiquetación refleja temas de manera más limpia, es decir, no se mezclan palabras como en el caso de cinco tópicos y no dividen los cúmulos como en siete tópicos.

b) Las etiquetas que se obtuvieron con los conjuntos de datos Corea, Salud, Siria, Suecia, Béisbol son:

Para 3 cúmulos las etiquetas son:

Tópico 0: Siria, probar, militar
Tópico 1: Béisbol, México, estigmatizar
Tópico 2: Suecia, luz, hogar

Para 4 cúmulos las etiquetas son:

Tópico 0: Suecia, luz, hogar
Tópico 1: Medicina, México, encuesta
Tópico 2: Siria, probar, militar
Tópico 3: León, béisbol, batear

Para 5 cúmulos las etiquetas son:

Tópico 0: Turquía, Siria, Israel
Tópico 1: Cáncer, medicina, México
Tópico 2: Irak, mandatario, miércoles
Tópico 3: Suecia, luz, hogar
Tópico 4: León, béisbol, batear

Para 6 cúmulos las etiquetas son:

Tópico 0: Irak, Siria, probar
Tópico 1: Turquía, Siria, Israel

Tópico 2: Cáncer, medicina, México
Tópico 3: México, médico, enfermedad
Tópico 4: Suecia, luz, hogar
Tópico 5: Buey, león, béisbol

Para 7 cúmulos las etiquetas son:

Tópico 0: Turquía, Siria, Israel
Tópico 1: México, encuesta, médico
Tópico 2: Césped, padre, pan
Tópico 3: Miércoles, probar, militar
Tópico 4: Medicina, México, estigmatizar
Tópico 5: Suecia, luz, hogar
Tópico 6: Buey, León, béisbol

Los conjuntos de datos para estas pruebas son cinco; dos hablan claramente de conflictos armados (Corea y Siria). A pesar de que los cúmulos comparten algunas palabras entre sí, las etiquetas indican claramente sus tópicos en el número de cúmulos naturales que es cinco. Para las configuraciones de tres y cuatro tópicos puede verse que los temas dominantes son Siria, Suecia y medicina. Para seis tópicos el conjunto de Siria se divide en dos cúmulos, nótese que se puede identificar gracias a la etiquetación. Para siete tópicos se crean dos cúmulos sobre béisbol, además, se crea otro conjunto que no se distingue claramente (Tópico 3), esto es porque no es una configuración de tópicos natural, nuevamente la etiquetación multi-palabra ayuda a identificar cuando el algoritmo LDA es forzado a entregar una configuración que no es semánticamente natural.

5.3 Pruebas para hallar el número de cúmulos naturales

A continuación, se presentan algunas tablas con pruebas realizadas al modelo para demostrar la forma de encontrar el número de cúmulos naturales creada en este trabajo.

En cada prueba se presentan los nombres de los archivos que contienen los conjuntos de datos (la cantidad de archivos empleados es el número de cúmulos naturales). Las Tablas 5.1,5.2,5.3,5.4 y 5.5 muestran el cálculo de los promedios de las compacidades o intra-distancias (primera columna); el promedio de las lejanías o inter-distancias (tercera columna) y el número X donde X= {2,3,4,5,6,7} de cúmulos en la columna central.

a)

Conjuntos de datos: Béisbol, Suecia, Salud, Universo

Cúmulos naturales: 4, Cúmulos por el modelo: 4

Tabla 5.1: Resultado de los cálculos de los promedios de las compacidades (columna 1) y lejanías (columna 3) usando diferentes 'x' (columna 2) para los conjuntos de datos: Béisbol, Suecia, Salud, Universo.

Compacidad c_x	x	**Lejanía ℓ_x**	c_x/ℓ_x
0.0786142429	2	352.7188756404	2.2288074E-04
0.0804723635	3	388.0515513906	2.0737545E-04
0.0753301039	4	439.9445709804	1.7122635E-04
0.0742185534	5	369.3469894553	2.0094533E-04
0.0790931109	6	373.8096751531	2.1158658E-04
0.0763198800	7	306.9209519519	2.4866298E-04

Sustituyendo los valores de la Tabla 5.1 en la definición 3, se obtiene:

$$K = \left[\arg\_min \left(\begin{matrix} \frac{0.0786142429}{352.7188756404}, \frac{0.0804723635}{388.0515513906}, \\ \frac{0.0753301039}{439.9445709804}, \frac{0.0742185534}{369.3469894553}, \\ \frac{0.0790931109}{373.8096751531}, \frac{0.0763198800}{306.9209519519} \end{matrix} \right) \right]$$

$$K = \left[\arg\_min \left(\begin{matrix} 2.2288074\mathrm{E}-04, 2.0737545\mathrm{E}-04, \\ 1.7122635\mathrm{E}-04, 2.0094533\mathrm{E}-04, \\ 02.1158658\mathrm{E}-04, 2.4866298\mathrm{E}-04 \end{matrix} \right) \right]$$

Ya que se comenzó de un X=2, la primera operación en la fórmula corresponde a la posición X=2, la segunda X=3, etc. Por lo tanto, para los valores de la Tabla 5.1 el K natural es: K=4. Véase graficación de los valores en Figura 5.1.

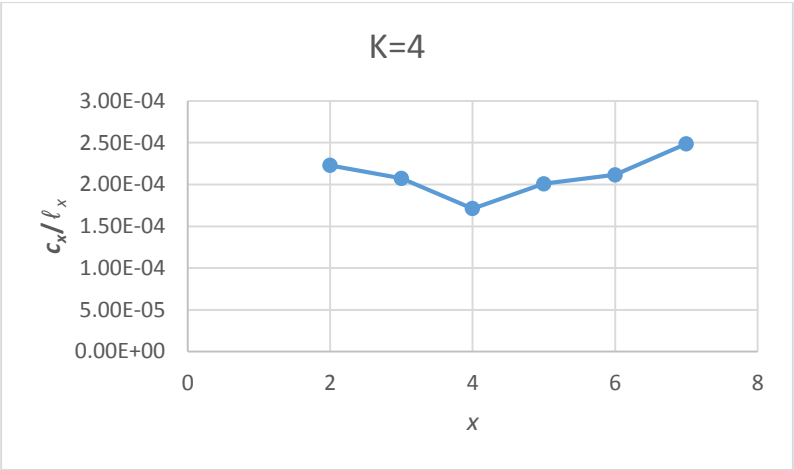


Figura 5.1: Gráfica de los valores registrados con los datos de la Tabla 5.1 usando la Definición 3.

Como puede verse empleando la Definición 3 se puede encontrar un número de cúmulos naturales que es similar a la forma de agruparlos por los humanos. La visualización de esta agrupación puede verse en la Figura 4.10 (sección anterior).

b)

Conjuntos de datos: Béisbol, Suecia, Salud, Universo, Louvre.

Cúmulos naturales: 5, Cúmulos por el modelo: 5

Tabla 5.2: Resultado de los cálculos de los promedios de las compacidades (columna 1) y lejanías (columna 3) usando diferentes 'X' (columna 2) para los conjuntos de datos: Béisbol, Suecia, Salud, Universo y Louvre.

Compacidad c_x	***x***	**Lejanía ℓ_x**	c_x/ℓ_x
0.0790816814	2	360.0330655855	2.1965116E-04
0.0803109502	3	357.1310956247	2.2487807E-04
0.0772891374	4	395.3550885980	1.9549296E-04
0.0756345618	5	402.8842017929	1.8773276E-04
0.0794700283	6	359.2302121295	2.2122312E-04
0.0767971263	7	375.1875158240	2.0468998E-04

$$K = \left[\arg\_min \left(\frac{0.0790816814}{360.0330655855}, \frac{0.0803109502}{357.1310956247}, \frac{0.0772891374}{395.3550885980}, \frac{0.0756345618}{402.8842017929}, \frac{0.0794700283}{359.2302121295}, \frac{0.0767971263}{375.1875158240} \right) \right]$$

$$K = \left[\arg\_min \begin{pmatrix} 2.1965116\text{E}-04, 2.2487807\text{E}-04, \\ 1.9549296\text{E}-04, 1.8773276\text{E}-04, \\ 2.2122312\text{E}-04 2.0468998\text{E}-04 \end{pmatrix} \right]$$

Ya que se comenzó de un X=2, la primera posición corresponde al X=2, la segunda X=3, etc. Por lo tanto, para los valores de la Tabla 5.2 el K natural es: K=5.

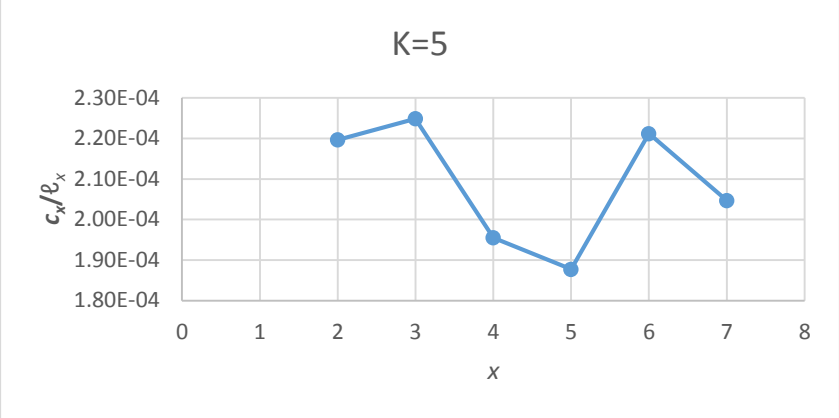


Figura 5.2: Gráfica de los valores registrados con los datos de la Tabla 5.2 usando la Definición 3.

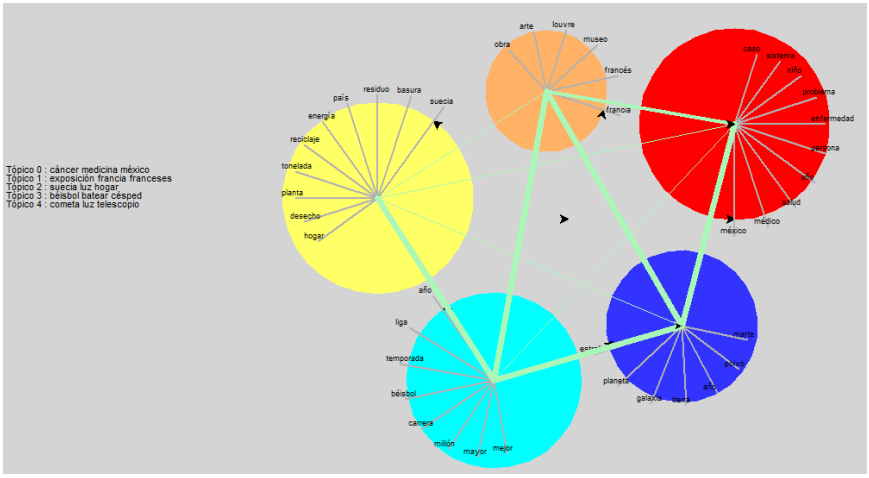


Figura 5.3: Visualización de los conjuntos de datos: Béisbol, Suecia, Salud, Universo, Louvre.

c)

Conjuntos de datos: Béisbol, Suecia y EPN.

Cúmulos naturales: 3, Cúmulos por el modelo: 3

Tabla 5.3: Resultado de los cálculos de los promedios de las compacidades (columna 1) y lejanías (columna 3) usando diferentes 'X' (columna 2) para los conjuntos de datos: Béisbol, Suecia y EPN.

Compacidad c_x	x	Lejanía ℓ_x	c_x/ℓ_x
0.0792850624	2	359.0519830415	2.2081778E-04
0.0764955347	3	391.3902271670	1.9544569E-04
0.0744735498	4	328.3673800173	2.2679948E-04
0.0793466616	5	279.4723087192	2.8391601E-04
0.0772648451	6	242.0541146020	3.1920484E-04
0.0779955452	7	229.9352717033	3.3920653E-04

$$K = \left[\arg\_min \left(\frac{0.0792850624}{359.0519830415}, \frac{0.0764955347}{391.3902271670}, \frac{0.0744735498}{328.3673800173}, \frac{0.0793466616}{279.4723087192}, \frac{0.0772648451}{242.0541146020}, \frac{0.0779955452}{229.9352717033} \right) \right]$$

$$K = \left[\arg\_min \left(\begin{matrix} 2.2081778\text{E}-04, 1.9544569\text{E}-04, \\ 2.2679948\text{E}-04, 2.8391601\text{E}-04, \\ 3.1920484\text{E}-04, 03.3920653\text{E}-04 \end{matrix} \right) \right]$$

Ya que se comenzó de un X=2, la primera posición corresponde al X=2, la segunda X=3, etc. Por lo tanto, para los valores de la Tabla 5.3 el K natural es: K=3.

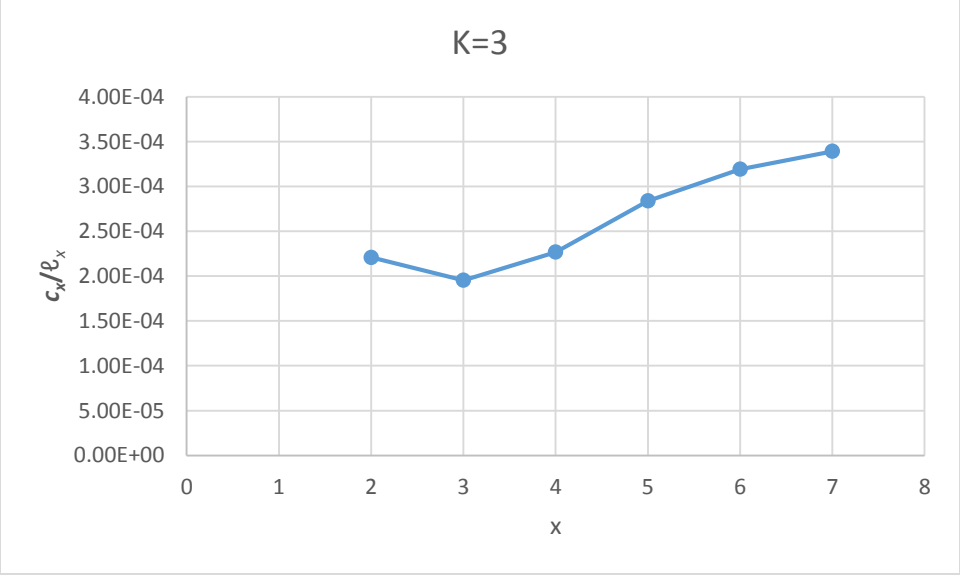


Figura 5.4: Gráfica de los valores registrados con los datos de la Tabla 5.3 usando la Definición 3.

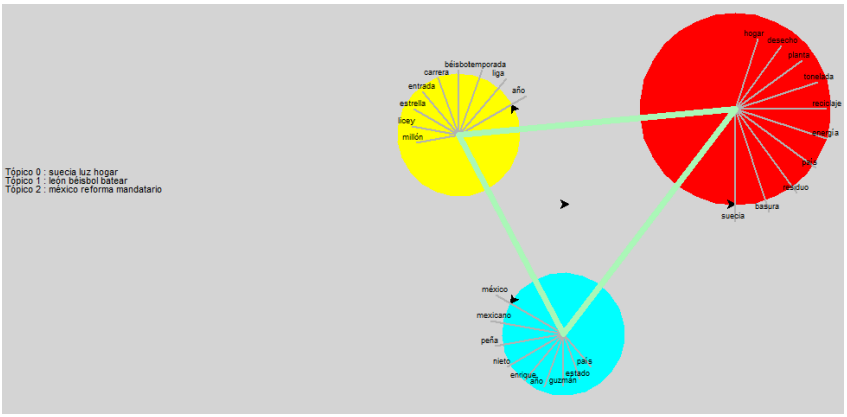


Figura 5.5: Visualización de los conjuntos de datos: Béisbol, Suecia y EPN

d)

Conjuntos de datos: Corea, EPN, Louvre, Salud, Universo.

Cúmulos naturales: 5, Cúmulos por el modelo: 5

Tabla 5.4: Resultado de los cálculos de los promedios de las compacidades (columna 1) y lejanías (columna 3) usando diferentes 'x' (columna 2) para los conjuntos de datos: Corea, EPN, Louvre, Salud, Universo.

Compacidad c_x	***x***	**Lejanía ℓ_x**	c_x/ℓ_x
0.0766949772	2	345.5623635094	2.2194251E-04
0.0821443623	3	313.0292633694	2.6241752E-04
0.0816420468	4	313.5356440343	2.6039160E-04
0.0757477259	5	407.7579277635	1.8576641E-04
0.0778212851	6	325.6165605342	2.3899671E-04
0.0768920226	7	375.0439589858	2.0502136E-04

$$K = \left[\arg\_min \left(\frac{0.0766949772}{345.5623635094}, \frac{0.0821443623}{313.0292633694}, \frac{0.0816420468}{313.5356440343}, \frac{0.0757477259}{407.7579277635}, \frac{0.0778212851}{325.6165605342}, \frac{0.0768920226}{375.0439589858} \right) \right]$$

$$K = \left[\arg\_min \begin{pmatrix} 2.2194251\text{E}-04, 2.6241752\text{E}-04, \\ 2.6039160\text{E}-04, 1.8576641\text{E}-04, \\ 2.3899671\text{E}-04, 2.0502136\text{E}-04 \end{pmatrix} \right]$$

Ya que se comenzó de un X=2, la primera posición corresponde al X=2, la segunda X=3, etc. Por lo tanto, para los valores de la Tabla 5.4 el K natural es: K=5.

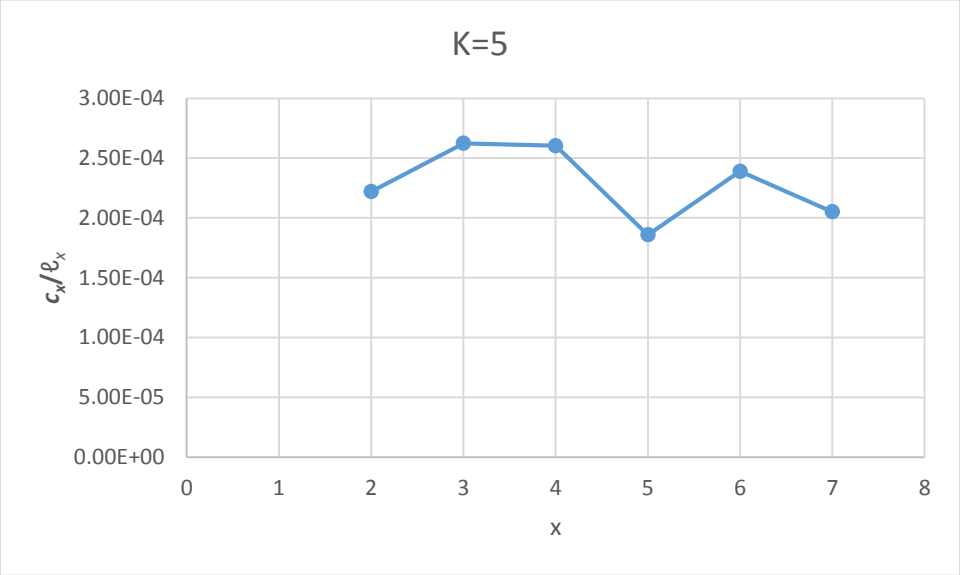


Figura 5.6: Gráfica de los valores registrados con los datos de la Tabla 5.4 usando la Definición 3.

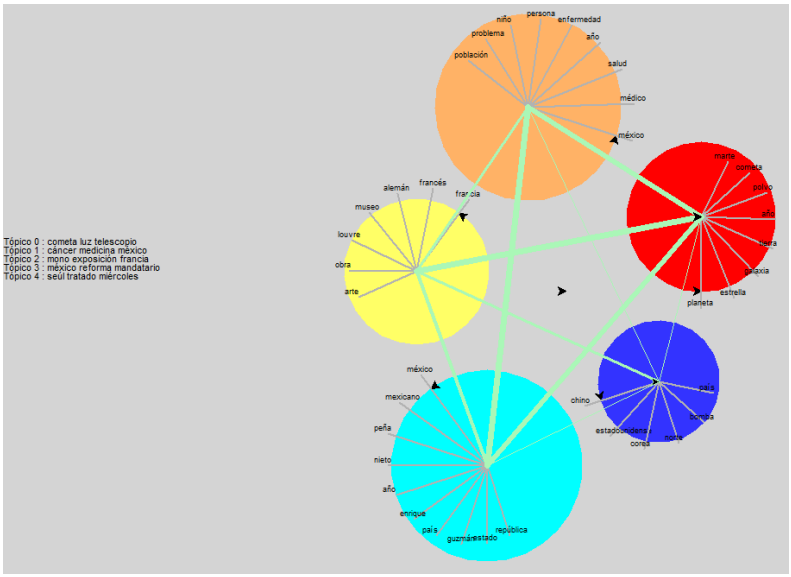


Figura 5.7: Visualización de los conjuntos de datos: Corea, EPN, Louvre, Salud, Universo.

e)

Conjuntos de datos: Corea, Louvre, Salud, Siria, Suecia, Universo

Cúmulos naturales: 6, Cúmulos por el modelo: 6

Tabla 5.5: Resultado de los cálculos de los promedios de las compacidades (columna 1) y lejanías (columna 3) usando diferentes 'X' (columna 2) para los conjuntos de datos: Corea, Louvre, Salud, Siria, Suecia, Universo.

Compacidad c_x	***x***	**Lejanía ℓ_x**	c_x/ℓ_x
0.0833208008	2	286.7580779472	2.9056130E-04
0.0821868389	3	317.6213018191	2.5875733E-04
0.0832829926	4	298.8961378770	2.7863522E-04
0.0817990332	5	359.5889325283	2.2747928E-04
0.0782834470	6	389.6978477722	2.0088242E-04
0.0825256435	7	314.8255560880	2.6213134E-04

$$K = \left[\arg\_min \left(\begin{matrix} \frac{0.0833208008}{286.7580779472}, \frac{0.0821868389}{317.6213018191}, \\ \frac{0.0832829926}{298.8961378770}, \frac{0.0817990332}{359.5889325283}, \\ \frac{0.0782834470}{389.6978477722}, \frac{0.0825256435}{314.8255560880} \end{matrix} \right) \right]$$

$$K = \left[\arg\_min \left(\begin{matrix} 2.9056130\mathrm{E}-04, 2.5875733\mathrm{E}-04, \\ 2.7863522\mathrm{E}-04, 2.2747928\mathrm{E}-04, \\ 2.0088242\mathrm{E}-04, 2.6213134\mathrm{E}-04 \end{matrix} \right) \right]$$

Ya que se comenzó de un X=2, la primera posición corresponde al X=2, la segunda X=3, etc. Por lo tanto, para los valores de la Tabla 5.5 el K natural es: K=6.

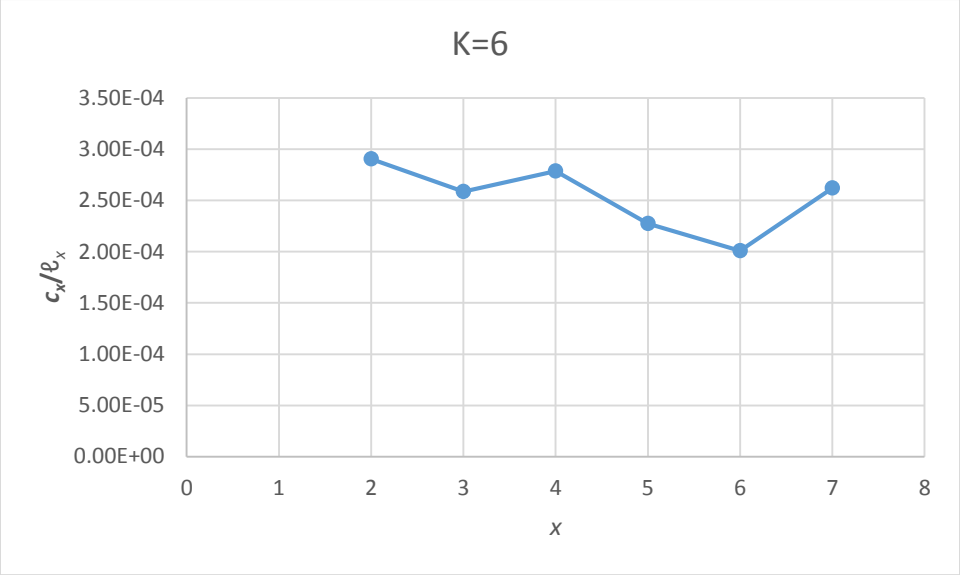


Figura 5.8: Gráfica de los valores registrados con los datos de la Tabla 5.5 usando la Definición 3.

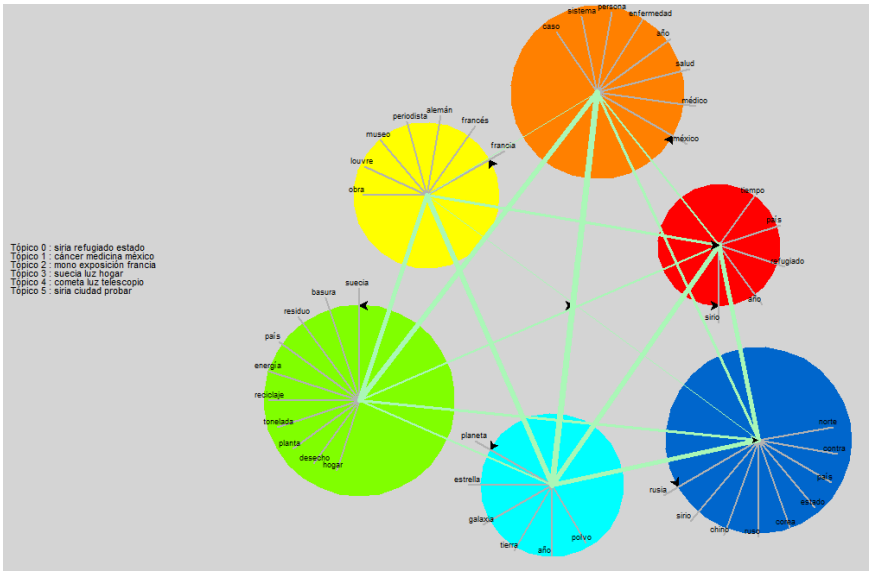


Figura 5.9: Visualización de los conjuntos de datos: Corea, Louvre, Salud, Siria, Suecia, Universo.

5.4 Pruebas con mayor cantidad de noticias

El documento con mayor cantidad de noticias es 100. Sin embargo, es difícil asegurar que en la vida real no surjan conjuntos de datos con mayor cantidad de documentos, es por ello, que en esta sección se presentan pruebas realizados a los mismos conjuntos de datos multiplicando su tamaño 10 y 100 veces más.

Prueba 1

Conjuntos de datos: Béisbol10, Suecia10, EPN10
Como puede verse en la Tabla 5.6, el número de cúmulos naturales encontrados por el modelo es K=3, es decir, sigue encontrando el número de cúmulos correctos a pesar de haber aumentado en 10 el número de noticias.

Tabla 5.6: Resultado de la ejecución para archivos de datos 10 veces más grandes a los originales, se usaron 1000 noticias de béisbol: 1000 de Suecia y 1000 de EPN.

Compacidades (promedio)	Cúmulos	Lejanías Naturales (promedio)
0.07878763464518564	2	347.3737948279482
0.0770493160199989	3	340.9546163177842
0.07944207977896868	4	260.97234851650535
0.07791295284234522	5	262.09696905194403
0.07962334205519976	6	283.5948737341321
0.07837154093321688	7	211.5994889330694

Prueba 2
Conjuntos de datos: Béisbol100, Suecia100, EPN100
Como puede verse en la Tabla 5.7, el número de cúmulos naturales encontrados por el modelo sigue siendo K=3, es decir, sigue encontrando el número de cúmulos correctos a pesar de haber aumentado por cien el número de noticias.

Tabla 5.7: Resultado de la ejecución para archivos de datos 100 veces más grandes a los originales, se usaron 10,000 noticias de béisbol: 10,000 de Suecia y 10,000 de EPN.

Compacidades (promedio)	Cúmulos	Lejanías Naturales (promedio)
0.07969504403420306	2	367.2060874711774
0.0770090602021512	3	354.76549457208245
0.0791019764947379	4	260.0477085041672
0.07895163350525712	5	302.82020357176395
0.07957901896788504	6	258.1566585510065
0.08047290932053344	7	234.27729325529532

Con las pruebas anteriores se comprueba que el modelo sigue encontrando el número de cúmulos naturales a pesar de escalar su tamaño.

5.5 Probando que existe más de una configuración de cúmulos naturales

En esta sección se muestra que existen más de una configuración de tópicos entendibles. Un ejemplo es: imaginar que se tienen los siguientes animales, perro,

lobo, gato y puma; ahora se les pide a diferentes personas que los agrupen sin ningún criterio específico esos animales. Tal vez la primera persona agrupe al g1= {perro y al gato} y g2= {lobo y puma}, dando como explicación que dos son animales domésticos y dos son salvajes. Puede que otra persona agrupe a los animales en g1= {perro y lobo} y g2= {gato y puma}, diciendo que perro y lobo son familiares y gato y puma lo son entre ellos. Como se lee, las dos clasificaciones anteriores son correctas, es decir, la clasificación es subjetiva (depende del observador).
Ahora una prueba real.

Conjuntos de datos: Ballenas, Tortugas, Salud, Gripe.
Número natural de cúmulos por el programa: 2
En esta prueba, el número natural de cúmulos que regresó el programa fue de 2 (Véase Tabla 5.13); antes de hacer la se sabía que el resultado sería 4, sin embargo, no lo fue. Lo anterior es como resultado de una prueba con trampa, ya que los archivos Ballenas y Tortugas contienen información muy parecida (de forma natural sin modificaciones); con los archivos gripa y medicina sucede algo similar. Se sabes que las tortugas son animales marinos al igual que las ballenas, ambos viven en los mares y/u océanos; en general, el contexto en el cuál se desenvuelven es similar. Por lo tanto, nuestro k=2 es una buena configuración de tópicos y tiene sentido.

Tabla 5.8: La tabla muestra que el modelo encuentra que el número de cúmulos naturales es 2.

Compacidad c_x	x	**Lejanía ℓ_x**	c_x/ℓ_x
0.0830898767	2	364.7440673652	2.2780323E-04
0.0830456393	3	287.4515381197	2.8890310E-04
0.0831497626	4	266.5199236625	3.1198329E-04
0.0790968073	5	288.5560735328	2.7411243E-04
0.0809968308	6	217.3238217803	3.7270112E-04
0.0800418890	7	240.5873290442	3.3269370E-04

$$K = \left[\arg\_min \left(\frac{0.0830898767}{364.7440673652}, \frac{\mathbf{0}0.0830456393}{287.4515381197}, \frac{0.0831497626}{266.5199236625}, \frac{0.0790968073}{288.5560735328}, \frac{0.0809968308}{217.3238217803}, \frac{0.0800418890}{240.5873290442} \right) \right]$$

$$K = \left[\arg\_min \left(\begin{matrix} 0.2.2780323\text{E}-04, 2.8890310\text{E}-04, \\ 3.1198329\text{E}-04, 2.7411243\text{E}-04, \\ 3.7270112\text{E}-04, 3.3269370\text{E}-04 \end{matrix} \right) \right]$$

Ya que se comenzó de un x=2, la primera posición corresponde al x=2, la segunda x=3, etc. Por lo tanto, para los valores de la Tabla 5.5 el K natural es: K=2.

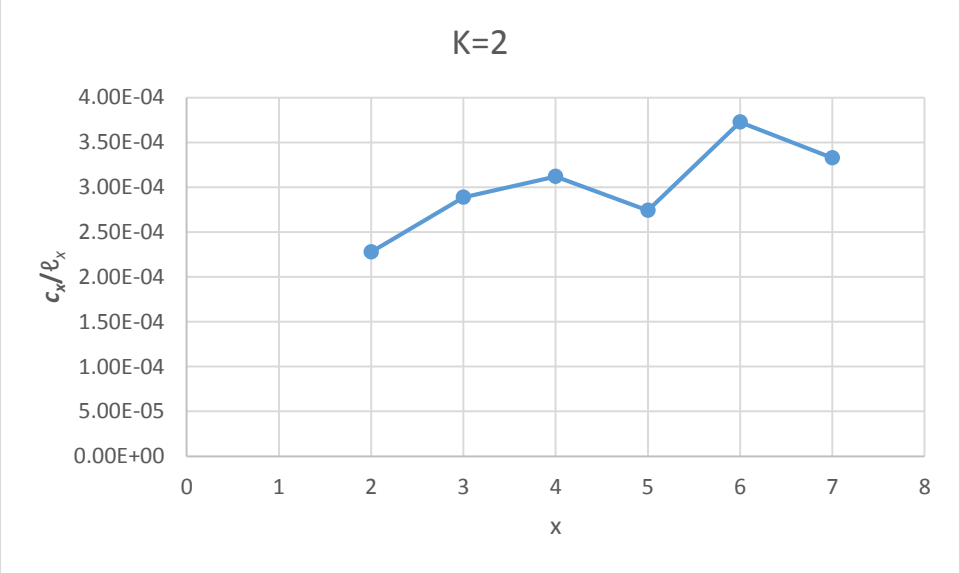


Figura 5.10: Gráfica de los valores registrados con los datos de la Tabla 5.8 usando la Definición 3.

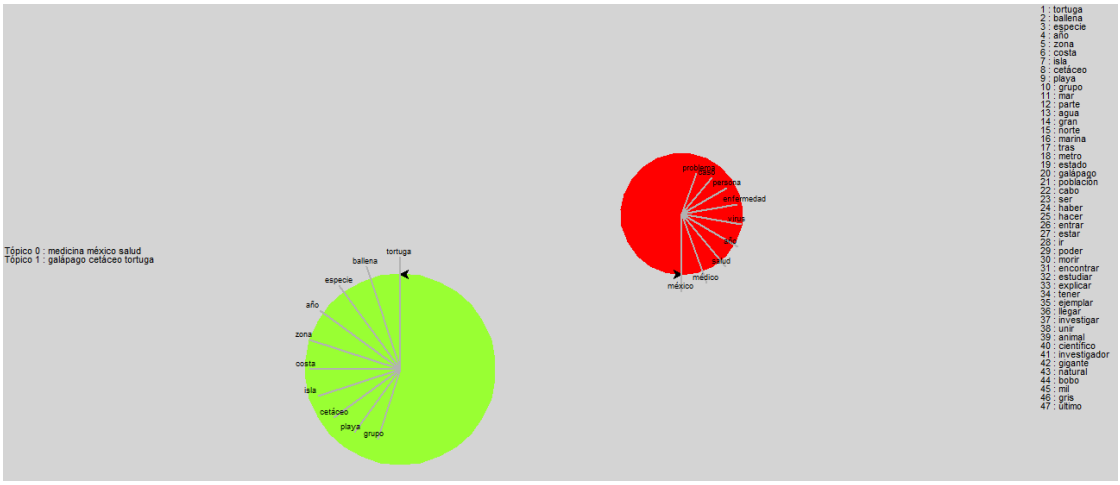


Figura 5.11: Visualización de los conjuntos de datos: Ballenas, Tortugas, Salud, Gripe.

5.6 Determinando el número de palabras que representan al cúmulo

En este apartado se muestra que porque se eligió que cincuenta fuera el número de palabras que representara cada cúmulo.

En la Tabla 5.9 se muestran los resultados de múltiples ejecuciones para encontrar los cúmulos naturales con cierta cantidad de palabras (el encabezado de las columnas es el número de palabras usado para la ejecución del programa), los cúmulos naturales para esos conjuntos de datos se resaltan en color amarillo. De la tabla puede observarse que, en 50 palabras por cúmulo, se descubre el número de cúmulos naturales con una menor profundidad de intersección cómo se hace en la sección 5.3, es decir, en 50 palabras por cúmulo el programa encontró en todos los casos el número natural de agrupaciones.

Tabla 5.9: Múltiples ejecuciones del cálculo de cúmulos naturales. Conjunto de datos 1: Béisbol, Salud, Universo, Suecia, Louvre. 2: Suecia, Universo, Salud, Béisbol. 3: EPN, Suecia, Béisbol.

Conjuntos de datos	20 Palabras	30 Palabras	40 Palabras	50 Palabras	60 Palabras	70 Palabras	80 Palabras
1	7	7	4	5	4	3	-
2	-	-	4	4	4	3	-
3	6	6	4	3	3	3	-

5.7 Elección de la medida de similitud

Se investigaron distintas medidas de similitud entre palabras; la Tabla 5.10 muestra un resumen sobre la forma en cómo se calculan y las desventajas para el modelo.

Tabla 5.10 Medidas de similitud entre palabras: Columna 1: nombre de la medida; columna 2, razón por la cual no es conveniente para el proyecto; columna 3, descripción del valor que regresa la función.

Medida de similitud	Razón por la cual se desecha	Valor regresado
Levenshtein [36]	No refleja la similitud semántica entre palabras	Cuenta el número mínimo de operaciones de inserción, borrado o substitución para transformar una palabra en otra.
Hamming [37]	No refleja la similitud semántica entre palabras	Aplica a palabras con la misma longitud y sólo admite operaciones de inserción.
Resnik´s measure [38]	Requiere que los synsets de las palabras que se comparan estén en la misma parte del discurso	Trata de cuantificar la cantidad de información entre dos conceptos: Basado en su ancestro más profundo (LCS Least Common Subsumer)
Lin's measure [59]	Requiere que los synsets de las palabras que se comparan estén en la misma parte del discurso	Es la relación del contenido de la información de LCS al contenido de la información a cada uno de los conceptos
Path measure [39]	Se lleva a la etapa de pruebas para definir su uso	Regresa un valor denotando la similitud basado en el camino más corto que conecta los sentidos de las palabras en la taxonomía (hiperónimo, hipónimo).
Wu and Palmer measure [40]	Se lleva a la etapa de pruebas para definir su uso	Se basa en la profundidad de los dos sentidos en la taxonomía y su LCS
Leacock and Chodorow measure [60]	Requiere que los Synsets de las palabras que se comparan estén en la misma parte del discurso	Usa la longitud del camino más corto y la máxima profundidad en la taxonomía de los conceptos comparados.
Jiang-Conrath [41]	Requiere que los Synsets de las palabras que se comparan estén en la misma parte del discurso	Regresa un valor de acuerdo con su ancestro en común más profundo y los dos Synsets de entrada

5.7.1 Elección de la medida de similitud

Con la ayuda de la Tabla 5.10 se sabe que las medidas *Path* y *Wu and Palmer* son candidatas a ser usadas en el proyecto para medir la similitud semántica entre palabras; sin embargo, debe verificarse si alguna tiene un mejor desempeño.
A continuación, muestran los modelos matemáticos de las medidas de similitud Path y Wu & Palmer.

- **Path**

$$sim_{path}(c_1, c_2) = 2 * deep_{max} - len(c_1, c_2)$$

Si $len(c_1, c_2)$ es 0, $sim_{path}(c_1, c_2)$ tiene su máximo valor $2 * deep_{max}$. Si $len(c_1, c_2)$ es $2 * deep_{max}$, $sim_{path}(c_1, c_2)$ obtiene su mínimo valor (0).
$0 \leq sim_{path}(c_1, c_2) \leq 2 * deep_{max}$

- **Wu & Palmer**

$$sim_{WP}(C_1, C_2) = \frac{2 * depth(lCS(c_1, c_2))}{len(c_1, c_2) + 2 * depth(lCS(c_1, c_2))}$$

Si $lCS(c_1, c_2)$ es la raíz, $depth(lCS(c_1, c_2)$=1, $sim_{WP}(C_1, C_2)$>0. Si los dos conceptos tienen el mismo sentido, están en el mismo nodo, $len(c_1, c_2)$=0, $sim_{WP}(C_1, C_2)$=1, de otra forma, 0<$depth(lCS(c_1, c_2)) < deep_{max}$.

La función de similitud Path considera el camino de menor longitud que conecta los sentidos de las palabras en la taxonomía; se toman en consideración las aristas de separación entre conceptos, mientras que Wu & Palmer se basa en el LCS y este, no necesariamente representa el camino más corto entre los sentidos de los conceptos. El LCS puede no cubrir el sentido de ambas palabras. Con esto se justifica la elección de la medida de similitud Path para los cálculos de esta tesis.

Adicional a lo anterior, para soportar de forma práctica la medida, se muestran dos ejercicios que soportan la elección.

Ejercicio 1

Conjuntos de datos: Béisbol, Salud, Universo, Suecia y EPN

La Tabla 5.11 muestra el promedio de las compacidades de los cúmulos para diferentes *K*; en verde se resaltan los valores más bajos, recordar que se busca que la compacidad tenga valores bajos. La similitud Path, acierta en el número de cúmulos naturales, la Wup no.

Tabla 5.11: Cálculo del promedio de las compacidades de los cúmulos para diferentes configuraciones de X.

Cúmulos X	Wup	Path
2	.2176	.0748
3	.2177	.0803
4	.2105	.07722
5	.2049	.0736251
6	.2017	.073767
7	.2082	.07684
8	.2053	.076156
9	.2166	.0792604
10	.2095	.07708

La Tabla 5.12 muestra el promedio de las lejanías naturales entre cúmulos, en verde están los valores más grandes, recordar que se desea que la lejanía entre cúmulos sea lo más grande posible. La similitud Path acierta en el número de cúmulos naturales, la Wup no.

Tabla 5.12: Cálculo del promedio de las lejanías naturales de los cúmulos para diferentes configuraciones de X.

Cúmulos X	Wup	Path
2	581.9029	364.235001438
3	641.700325252	398.123187916
4	616.497853097	388.518510555
5	610.234307611	400.666080794
6	586.136502479	387.970628902
7	490.053104552	324.898696573

8	503.961850609	341.487261282
9	474.86677569	305.451791179
10	415.971856235	270.301158465

De las tablas anteriores se concluye que usando la similitud Path el promedio de las distancias entre cúmulos obtiene valores más grandes, es lo deseado. No obstante, una prueba más a continuación.

Ejercicio 2

Conjuntos de datos: Béisbol, Salud, Universo, Suecia, Louvre y EPN

Para esta prueba ninguna de las dos similitudes acierta en el número de cúmulos naturales que es 6, sin embargo, la similitud Path se acerca más que la similitud Wup. En la Tabla 5.13 aparecen los resultados de la ejecución del modelo.

Tabla 5.13: Cálculo del promedio de las compacidades de los cúmulos para diferentes configuraciones de X.

Cúmulos X	Path	Wup
2	0.0777400278264	0.209689861588
3	0.0771450416179	0.205022246472
4	0.0806943657409	0.22138751214
5	0.0743131822029	0.2233629436
6	0.0811252634462	0.212067016174
7	0.0759204583831	0.213244340179
8	.0760496455129	0.209827669283
9	0.0776052587531	0.214201393884
10	0.0755380608094	0.20779332513

Para este ejemplo la similitud Path obtuvo la separación entre cúmulos más grande (Véase la Tabla 5.14 para ver los resultados de la ejecución).

Tabla 5.14: Cálculo del promedio de las lejanías naturales de los cúmulos para diferentes configuraciones de X.

Cúmulos X	Path	Wup
2	392.228453989	615.306241595
3	380.023993854	618.294849211
4	364.854508344	610.752219061
5	319.961136914	554.972074467
6	396.804109805	593.811856349
7	325.688818135	512.456340888

8	314.616221468	492.91894628
9	278.448694961	439.163146704
10	310.230706	472.290482558

Con base en las mediciones anteriores pudo verse que la similitud *Path* proporciona valores más más grandes en las lejanías naturales y valores más pequeños en las compacidades de los cúmulos; por lo tanto, es la medida empleada para calcular las similitudes entre palabras de este trabajo.

5.8 Leyes de texto

En esta sección se presentan las leyes del texto que como es de esperarse comprobarse con el modelo.

5.8.1 Ley de Zipf

En esta prueba se visualiza la Ley de Zipf para los conjuntos de datos: Universo, Béisbol y EPN.

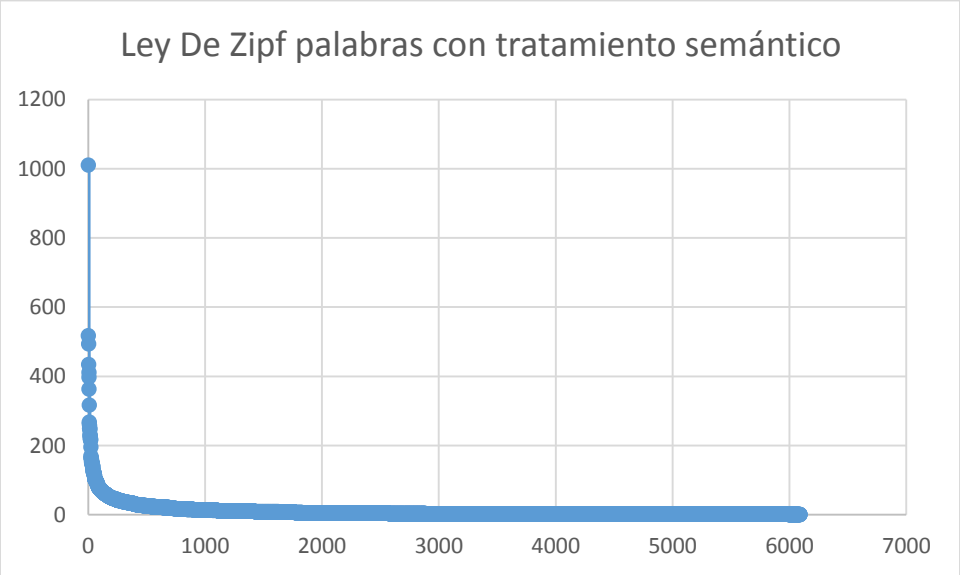


Figura 5.12: Ley de Zipf para los conjuntos de datos Universo, Béisbol y EPN.

5.8.2 Ley de Heaps

En esta prueba se visualiza la Ley de Heaps para los conjuntos de datos: Universo, Béisbol y EPN.

Para la ecuación de Heaps $v = kT^b$ se obtuvo que para un k=16 y una b=.49; el vocabulario de los documentos sigue el comportamiento de la Figura 5.8. La Tabla 5.15, en la primera columna se muestra el porcentaje de los documentos y en la columna dos la cantidad de noticias en valor numérico. La columna tres muestra el total de palabras que contienen las noticias. En la columna cuatro se muestra el tamaño del vocabulario real y en la quinta columna se muestra el vocabulario estimado con la ecuación de Heaps.

Tabla 5.15: Se muestra el tamaño el vocabulario estimado con de la Ley de Heaps, en la Figura 5.8 puede verse el comportamiento de la curva.

Porcentaje de documentos	Noticias	Palabas en los documentos	Tamaño vocabulario	Tamaño vocabulario Ley de Heaps
0	0/0/0	0	0	0
10	6/10/10	11094	1356	1535.3704
20	13/20/20	22787	2178	2184.6742
30	20/30/30	34480	2689	2676.2577
40	26/40/40	45574	3001	3068.2545
50	33/50/50	57267	3435	3431.5723
60	39/60/60	68361	4088	3742.6231
70	46/70/70	80054	4383	4043.6882
80	52/80/80	91148	4882	4309.1943
90	59/90/90	102841	5486	4571.7393
100	66/100/100	114534	6090	4819.4541

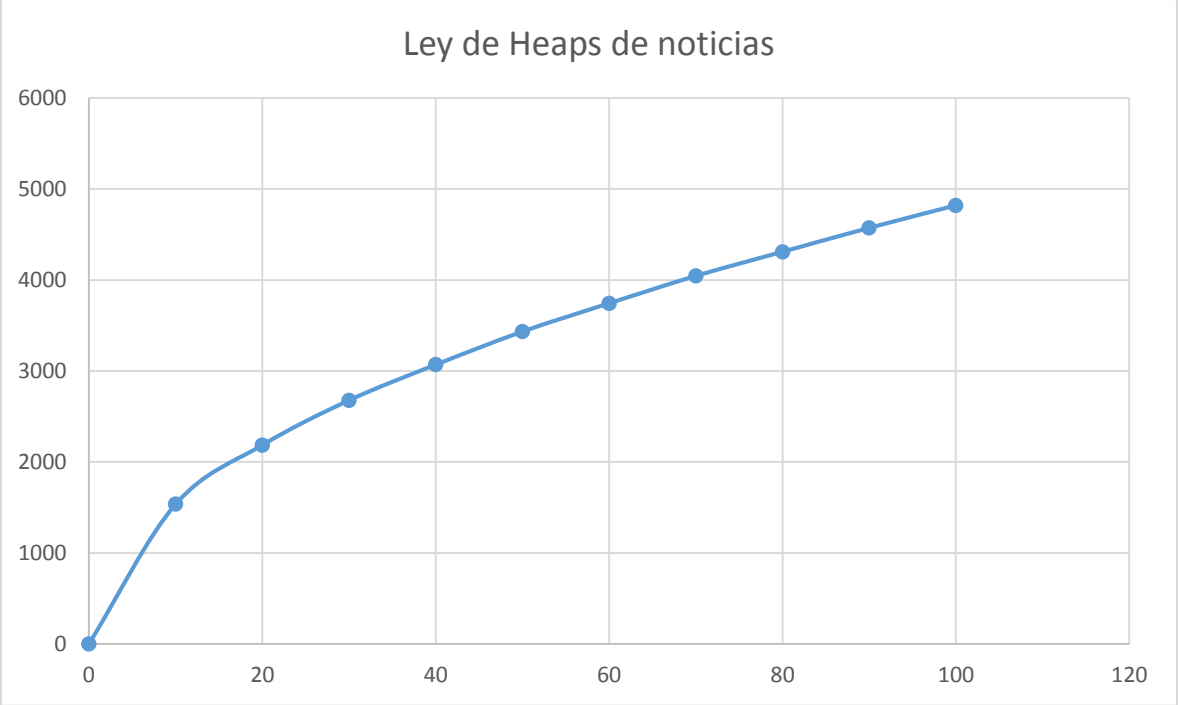


Figura 5.13: Curva característica de la Ley de Heaps para los conjuntos de datos Universo, Béisbol y EPN.

6. Conclusiones y contribuciones

6.1 Conclusiones

En esta tesis se diseñó, implementó y comprobó un modelo creado para detectar el número de cúmulos naturales en una clasificación no supervisada de texto. Los documentos que recibe el modelo como entrada son conjuntos de noticias provenientes de la prensa digital, es decir, es información no estructurada.

El número de cúmulos naturales, se encontró forzando al algoritmo LDA que es un clasificador no supervisado a comportarse como un clasificador supervisado; para ello se empleó una taxonomía conocida como Wordnet para ayudar a conocer la relación semántica entre palabras y con ello calcular la compacidad de cada cúmulo al igual que la lejanía natural que existe entre cúmulos.

Cada agrupación de la clasificación es etiquetada usando las relaciones semánticas entre palabras, para ello nuevamente se emplea Wordnet. Esta etiquetación emplea las palabras con mayor jerarquía en la taxonomía.

Finalmente, se desarrolló una visualización capaz de mostrar múltiples propiedades de la clasificación como: la cantidad de documentos asignados a cada cúmulo, la cantidad de palabras que comparten las agrupaciones, las palabras con mayor importancia de la agrupación y la lista completa de las palabras por cúmulo.

6.2 Contribuciones

Las contribuciones importantes de este trabajo son: (se copió otra vez, estaba al principio)

(d) Se ha encontrado un modelo eficaz para convertir un clasificador no supervisado en uno supervisado, sin conocer de antemano las clases que se hallarán. Con éste es posible agrupar un conjunto de documentos (noticias en español de la prensa diaria) según los temas que abordan. El algoritmo se comporta

como un clasificador supervisado porque a cada grupo (clase) le da una etiqueta (una a tres palabras en español), que denota el tema del que habla el grupo.

(e) Estos grupos concuerdan razonablemente con los que las personas producirían, por lo que se dice que el algoritmo *extrae* la semántica de las noticias: las agrupa por los temas principales que abordan.

(f) El algoritmo maximiza la distancia semántica entre los conceptos que pertenecen a un mismo grupo, y maximiza la distancia entre grupos distintos.

REFERENCIAS

[1] T. Wei, Y. Lu, H. Chang, Q. Zhou, and X. Bao, "A semantic approach for text clustering using WordNet and lexical chains," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 2264–2275, 2015.

[2] A. Guzmán, "Finding the main themes in a spanish document," *Expert Syst. Appl.*, vol. 14, no. 1–2, pp. 139–148, 1998.

[3] S. Levachkine, A. Guzmán-Arenas, and V. P. De Gyves, "The Semantics of Confusion in Hierarchies : Theory and Practice," *Contrib. to ICCS 05 13th Int. Conf. cenceptual Struct. Common Semant. Shar. Knowl.*, no. Cic, 2005.

[4] C. Lipizzi, L. Iandoli, and J. E. Ramirez Marquez, "Extracting and evaluating conversational patterns in social media: A socio-semantic analysis of customers' reactions to the launch of new products using Twitter streams," *Int. J. Inf. Manage.*, vol. 35, no. 4, pp. 490–503, 2015.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *J. Mach. Learn. Res.*, vol. 3, no. 4–5, pp. 993–1022, 2012.

[6] G. E. Noel and G. L. Peterson, "Applicability of Latent Dirichlet Allocation to multi-disk search," *Digit. Investig.*, vol. 11, no. 1, pp. 43–56, 2014.

[7] C. Vicient and A. Moreno, "Unsupervised topic discovery in micro-blogging networks," *Expert Syst. Appl.*, vol. 42, no. 17–18, pp. 6472–6485, 2015.

[8] M. Kar, S. Nunes, and C. Ribeiro, "Summarization of changes in dynamic text collections using Latent Dirichlet Allocation model," *Inf. Process. Manag.*, vol. 51, no. 6, pp. 809–833, 2014.

[9] Z. Li, J. Li, Y. Liao, S. Wen, and J. Tang, "Labeling clusters from both linguistic and statistical perspectives: A hybrid approach," *Knowledge-Based Syst.*, vol. 76, pp. 219–227, 2015.

[10] A. Nürnberger and M. Detyniecki, "Externally growing self-organizing maps and its application to e-mail database visualization and exploration," *Appl. Soft Comput. J.*, vol. 6, no. 4, pp. 357–371, 2006.

[11] M. Yuan and Y. Shi, "Text Clustering Based on a Divide and Merge Strategy," *Procedia Comput. Sci.*, vol. 55, no. Itqm, pp. 825–832, 2015.

[12] W. Hang, F. Chung, and S. Wang, "Transfer affinity propagation-based clustering," *Inf. Sci. (Ny).*, vol. 348, pp. 337–356, 2016.

[13] J. Liang, X. Zhao, D. Li, F. Cao, and C. Dang, "Determining the number of clusters using information entropy for mixed data," *Pattern Recognit.*, vol. 45, no. 6, pp. 2251–2265, 2012.

[14] S. Singaravelu, A. Sherin, and S. Savitha, "Agglomerative Fuzzy K-Means Clustering Algorithm," pp. 16–20, 2013.

[15] C. a Sugar and G. M. James, "Finding the number of clusters in a data set : {A}n information theoretic approach," *J. Am. Stat. Assoc.*, no. 1998, pp. 750–763, 2003.

[16] M. Aghagolzadeh, B. N. Araabi, and I. Processing, "the Clusters Dataset Theoretic Algorithm," no. 2, pp. 1336–1339.

[17] K. Chen and L. Lui, "The' Best K' for entropy-based categorical data clustering," *SSDBM'2005 Proc. 17th Int. Conf. Sci. Stat. database Manag.*, pp. 253–262, 2005.

[18] H. Yan, K. Chen, L. Liu, and J. Bae, *Determining the best K for clustering transactional datasets: A coverage density-based approach*, vol. 68, no. 1. 2009.

[19] L. Bai, J. Liang, and C. Dang, "An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data," *Knowledge-Based Syst.*, vol. 24, no. 6, pp. 785–795, 2011.

[20] H. Sun, S. Wang, and Q. Jiang, "FCM-based model selection algorithms for determining the number of clusters," *Pattern Recognit.*, vol. 37, no. 10, pp. 2027–2037, 2004.

[21] S. Saha and S. Bandyopadhyay, "A symmetry based multiobjective clustering technique for automatic evolution of clusters," *Pattern Recognit.*, vol. 43, no. 3, pp. 738–751, 2010.

[22] C. Bouras and V. Tsogkas, "A clustering technique for news articles using WordNet," *Knowledge-Based Syst.*, vol. 36, pp. 115–128, 2012.

[23] M. T. Hassan, A. Karim, J. B. Kim, and M. Jeon, "CDIM: Document Clustering by Discrimination Information Maximization," *Inf. Sci. (Ny).*, vol. 316, pp. 87–106, 2015.

[24] J. De Knijff, F. Frasincar, and F. Hogenboom, "Domain taxonomy learning from text: The subsumption method versus hierarchical clustering," *Data Knowl. Eng.*, vol. 83, pp. 54–69, 2013.

[25] W. Song, J. Z. Liang, and S. C. Park, "Fuzzy control GA with a novel hybrid semantic similarity strategy for text clustering," *Inf. Sci. (Ny).*, vol. 273, pp. 156–170, 2014.

[26] Y. Li, S. M. Chung, and J. D. Holt, “Text document clustering based on frequent word meaning sequences,” *Data Knowl. Eng.*, vol. 64, no. 1, pp. 381–404, 2008.
[27] J. Zhang, *The Information Retrieval Series*. 2008.
[28] G. Sidorov, “Non-linear construction of n-grams in computational linguistics: syntactic, filtered, and generalized n-grams,” 2013.
[29] D. P. Pancho, J. M. Alonso, O. Cordon, A. Quirin, and L. Magdalena, “Fingrams: Visual representations of fuzzy rule-based inference for expert analysis of comprehensibility,” *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 6, pp. 1133–1149, 2013.
[30] J. Ahn, “What you see is what you search: adaptive visual search framework for the web,” *19th Int. Conf. World wide web*, pp. 1049–1050, 2010.
[31] L. Padró, “FreeLing User Manual,” no. October, p. 95, 2013.
[32] L. Meng, R. Huang, and J. Gu, “A Review of Semantic Similarity Measures in WordNet,” *Int. J. Hybrid Inf. Technol.*, vol. 6, no. 1, pp. 1–12, 2013.
[33] J. H. Zar, “Statistical analysis.,” p. 663, 1999.
[34] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, “Introduction to wordnet: An on-line lexical database,” *Int. J. Lexicogr.*, vol. 3, no. 4, pp. 235–244, 1990.
[35] R. Rada, H. Mili, E. Bicknell, and M. Bletner, “Development and applications of a metric on semantic netsb,” *Trans. Syst. Man Cybern.*, vol. 1, pp. 17–30, 1989.
[36] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” *Soviet Physics Doklady*, vol. 10, no. 8. pp. 707–710, 1966.
[37] R. W. Hamming, “Error Detection and error Correcting Codes,” *Teleph. Telegr. Co.*, p. 14, 1950.
[38] P. Resnik, “Using Information Content to Evaluate Semantic Similarity in a Taxonomy,” *roceedings 14th Int. Jt. Conf. Artif. Intell. - Vol. 1 - IJCAI’95*, vol. 1, p. 6, 1995.
[39] H. Bulskov, R. Knappe, and T. Andreasen, “On measuring similarity for conceptual querying,” *Flex. Query Answering Syst.*, pp. 100–111, 2002.
[40] Z. Wu and M. Palmer, “Verb semantics and lexical selection,” *32nd Annu. Meet. Assoc. Comput. Linguist.*, pp. 133–138, 1994.
[41] J. J. Jiang and D. W. Conrath, “Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy,” *Proc. Int. Conf. Res. Comput. Linguist.*, no. Rocling X, pp. 19–33, 1997.
[42] R. Willi, L. P. Coelho, “Building Machine Learning Systems with Python”, 2013.
[43] S. N. Galicia, A. Gelbukh, “Investigaciones en análisis sintáctico para el español”, 2007.
[44] A. Ramírez, “Ajuste de parámetros de un clasificador asociativo aplicando metaheurísticas”, 2015.
[45] ITU, “Informe sobre Medición de la Sociedad de la Información”, 2015.
[46] B. MIrkin, “Choosing the number of clusters”, Data Mining and Knowledge Discovery Volume 1, Issue 3, pages 252–260, May/June 2011.
[47] N. Indurkhya and F. Damerau, “HandBook of Natural Language Processing”, second edition, 2010.
[48] A. Gelbukh, G. Sidorov, “Procesamiento automático del español con enfoque en recursos léxicos grandes”, segunda edición, 2010.
[49] E. Ovchinnikova, “Integration of World Knowledge for Natural Language Understanding”, 2012.
[50] J. Leskovec, A. Rajaraman y J. D. Ullman, “Mining of massive data sets”, 2014.
[51] I. Witten, E. Frank y M. Hall, “Data Mining practical machine learning techniques”, third edition, 2011.
[52] M. Hearst, “Text data mining: Issues, techniques, and the relationship to information access”, 1997.
[53] Stephen Marsland , “Machine Learning an Algorithmic perspective”, 2009.
[54] O. Duda, P. Hart, D Stork, “Pattern Classification”, 2000.
[55] M. Guerrero, “Sistema de visualización de la información de tópicos más importantes generados en medios sociales”, CIC IPN, 2015.
[56] K. Zipf, “Human behavior and the principle of least effort”. Oxford, England: Addison-Wesley Press (1949).
[57] H. Heaps, “Information Retrieval: computational and theoretical aspects”. Academic Press, Inc. Orna-lando, FL, USA (1978).
[58] M. Ruseell, “Mining the social Web”, ISBN: 978-1-449-38834-8, 2011.
[59] D. Lin, “An information-theoretic definition of similarity”, Proceedings of the 15th International Con. ference on Machine Learning, (1998) July 24-27; Madison, Wisconsin, USA.
[60] C. Leacock and M. Chodorow, Combining Local Context and WordNet Similarity for Word Sense Identification, WordNet: An Electronic Lexical Database, MIT Press, (1998), pp. 265-283.

[61] http://www.rss.nom.es/
[62] http://nlp.lsi.upc.edu/freeling/
[63] http://adimen.si.ehu.es/web/MCR
[64] http://www.cnnexpansion.com/
[65] http://www.reforma.com/
[66] http://www.eluniversal.com.mx/
[67] http://www.20minutos.com.mx/
[68] http://noticiasdelaciencia.com/
[69] http://www.bbc.com/
[70] http://www.conacytprensa.mx/
[71] http://www.who.int/en/
[72] https://code.google.com/p/stop-words/
[73] https://code.google.com/p/diccionariosinonimos/
[74] https://pypi.python.org/pypi
[75] https://es.wiktionary.org/wiki/Wikcionario:Portada
[76] http://www.nltk.org/howto/wordnet.html