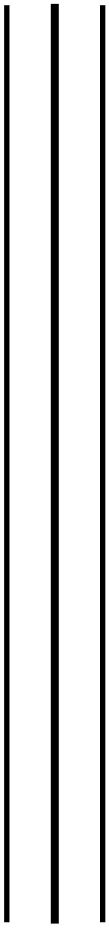
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS



Minado de tópicos usando min-Hashing en Tesis UNAM y su visualización

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Matemático

P R E S E N T A :

Ernesto López Cacho

DIRECTOR DE TESIS:
Dr. Ivan Vladimir Meza Ruiz

Ciudad Universitaria, CD.MX. 2021

Índice

Capítulo 1: Introducción....3
1.1. Objetivos....5
1.2. Metas....5
1.3. Organización de la tesis....6
Capítulo 2: Antecedentes....7
2.1. Palabras clave....7
2.2. Minería de textos....10
2.3. Modelado de tópicos....10
2.4. Métodos de modelado de tópicos....12
2.4.1. Latent Dirichlet Allocation....15
2.5. Non-negative Matrix Factorization....22
2.6. Tópicos en el tiempo....26
2.7. Tesis UNAM....31
Capítulo 3: Minado de tópicos: SWMH....32
3.1. Conjuntos semejantes....32
3.2. Funciones Hash....33
3.2.1. *Locality-sensitive hashing*....34
3.2.2. Min-Hashing....34
3.2.3. Min-Hashing para búsqueda de pares de conjuntos similares....35
3.3. *Sampled Min-hashing* para minado de tópicos....38
3.4. Etapa I: Partición....39
3.4.1. Selección de parámetros....42
3.5. Etapa II: Agrupamiento....43
3.6. Samples Weighed minhashing....44
3.7. Visualización minHash....46
Capítulo 4: Desarrollo y Experimentación....47
4.1. Adquisición y preprocesamiento de datos....48
4.2. Procesamiento de datos con *SWMH*....50
4.2.1. Tópicos como cuentas....51
4.2.2. Uso de *stopwords*....52
4.2.3. Umbral de correlación....52
4.3. Métricas....53
4.4. Experimentos....55
4.4.1. Selección de modelo....55
4.4.2. Resultados de *r3-l400*....58
Capítulo 5: Conclusiones....69
Capítulo 6: Anexo: Tópicos completos....72
Capítulo 7: Bibliografía....80

Capítulo 1: Introducción.

En la actualidad, gracias al avance agigantado en todos los aspectos de las tecnologías web, redes sociales y digitalización, nos encontramos con una generación inmensa de información día a día. Mucha de esta información está en forma de texto y conforma acerbos enormes de documentos. Al hablar de documentos, una idea que se viene a la mente son las colecciones que se generan a partir de revistas o periódicos digitales y las piezas literarias en las bibliotecas modernas. No obstante, existen otras fuentes que generan una inmensa cantidad de información, por ejemplo, las redes sociales. Si se considera cada mensaje o cada *post* en ellas como un documento es perceptible lo abrumador que puede ser. Para darse una idea, la estadística de Twitter reporta que se emiten 6,000 *tweets* cada segundo, dando un aproximado de 500 millones de *tweets* al día[1].

Con toda esta generación de información resulta inmediata la pregunta: ¿De qué hablan tantos documentos? Es evidente que sería muy complicado que un humano o grupo de humanos leyeran todos estos documentos y generaran reportes sobre los temas que tocan. Para esta labor se han desarrollado métodos computacionales que ayudan a encontrar estos temas usando distintas estrategias matemáticas. A los temas encontrados con estos métodos se les asigna el nombre técnico de tópicos.

En un análisis de tópicos se pueden encontrar las temáticas que se tocan en toda la colección y con eso entender líneas generales y patrones de comportamiento. Por ejemplo, en redes sociales se puede observar cuándo surge un asunto y seguirlo hasta que desaparece. Es interesante para puntos de vista políticos (campañas) y sociales pero también podría serlo en ámbitos más grandes de inteligencia y seguridad nacional.

En este trabajo se hará un estudio de tópicos sobre las tesis que se han hecho en la UNAM en el área de cómputo. Con esto, se observará cuáles han sido los temas que se han tratado

y cuáles han sido más llamativos para los alumnos o la comunidad en general y se mostrará su presencia a lo largo del tiempo.

La presente investigación es relevante porque las tesis realizadas pueden ser un muestreo de los campos de domino e interés entre la comunidad universitaria; entonces, al hacer este estudio se puede identificar los temas que han abarcado esfuerzo dentro de la universidad y observar cómo estos cambian en el tiempo. Incluso se podrían hacer comparaciones con otras universidades de México o si se encuentra la forma de sortear la barrera del lenguaje también se podría comparar con universidades del mundo. Otro enfoque interesante es que la UNAM es una de las mayores fuentes de conocimiento para México, entonces, al identificar los temas de trabajo se puede tener una idea de en qué líneas ha invertido esfuerzo el país. No obstante, realizar un análisis de este tipo tiene complicaciones en varios niveles, desde los técnicos y metodológicos hasta los sociales y burocráticos.

Al igual que en todo trabajo de análisis de datos el primer problema, y el más importante, es tener los datos. En este caso, obtener las tesis representa un esfuerzo adicional ya que, a pesar que la consulta de los documentos es abierta y publica a través de internet, no existe una manera directa de obtener más de una a la vez. Por ello, resultó complejo ya que se tuvieron que implementar programas con cierto nivel de complejidad los cuales en conjunto navegan la página de tesis UNAM y de forma automática descargan el subconjunto de las tesis en cómputo para formar la base de datos de trabajo. En principio, esta base de datos debería de ser de más fácil descarga y sin el problema adicional que se tuvo que superar para desbloquear los documentos y que fueran manejables.

El siguiente reto importante está relacionado con la calidad de los datos. En este caso, las tesis anteriores al año 2009 están digitalizadas mediante un escáner y un software de reconocimiento de texto, el cual convierte las imágenes a texto digital. Ésta es una fuente de error seria ya que si este programa comete errores en la interpretación de los caracteres ─que es común que ocurra─ entonces los documentos tienen deficiencias.

En el aspecto técnico, también hay que mencionar que tanto el equipo de cómputo como la conexión de red tienen que tener ciertas características, para poder descargar, almacenar, manejar la base de datos principal y las auxiliares así como para ejecutar los algoritmos que descubran los tópicos en un tiempo razonable para realizar varios experimentos. En este trabajo se utilizó un servidor para experimentación en el IIMAS de la UNAM para que con su poder de cómputo se agilizaran todos los procesos.

1.1. Objetivos

El **objetivo principal** del trabajo es identificar, en las tesis de computación, los temas que han tenido mayor interés año con año y analizar si estos permanecen por periodos largos de tiempo o si se encuentra algún comportamiento periódico.

Para lograr el objetivo principal hay una serie de **objetivos particulares** que se tendrán que alcanzar primero los cuales son:

1. Obtener un corpus adecuado con todas las tesis de computación.
2. Implementar un sistema adecuado para obtener los tópicos.
3. Realizar la minería de tópicos con distintos parámetros.
4. Visualizar y analizar las relaciones temporales de los tópicos.

1.2. Metas

Cada uno de estos objetivos particulares podrá ser alcanzando cumpliendo ciertas **metas**.

1. Implementar toda la infraestructura necesaria para el minado de tópicos (Hardware, Sistemas básicos y la base de datos).
2. Desarrollar el software necesario para las necesidades específicas.
3. Realizar experimentos con distintos parámetros para el modelado de tópicos.

4. Desarrollar una visualización y realizar análisis sobre los modelos.

1.3. Organización de la tesis

La estructura de la tesis se divide en 4 capítulos adicionales como sigue:

- En el Capítulo 2: se presentan los antecedentes que permiten entender los conceptos básicos que se utilizan en este tipo de análisis.
- En el Capítulo 3: se detalla *Sampled MinHashing*, que es la metodología particular que se utiliza en este caso para encontrar los tópicos.
- En el Capítulo 4: se muestra el camino que se siguió para realizar los experimentos y los resultados que se obtuvieron.
- Finalmente, en el Capítulo 5: se presentan las conclusiones a las que se llegaron.

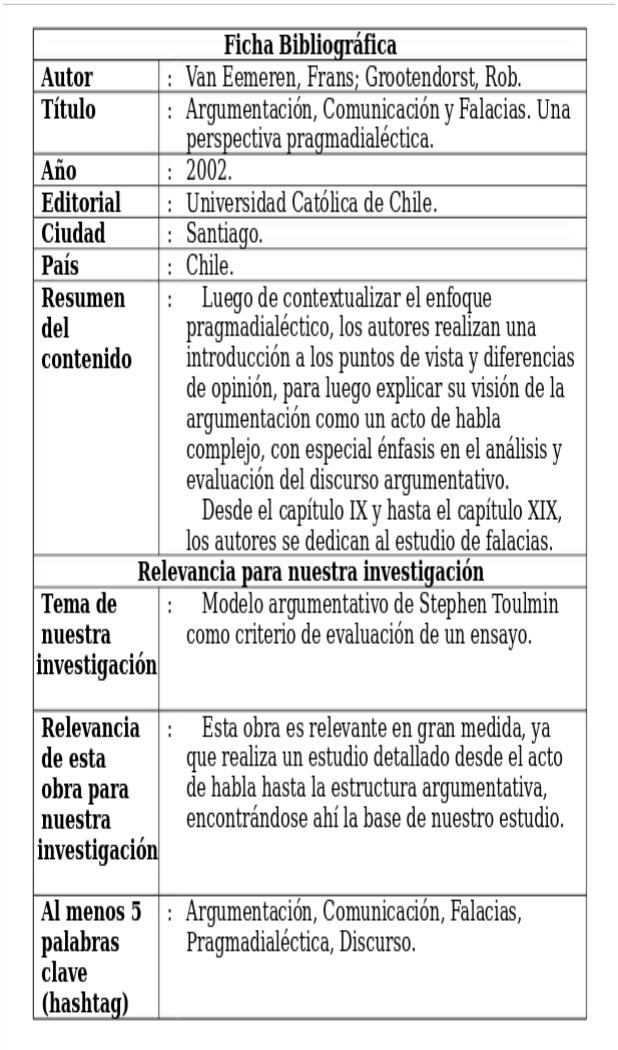
Capítulo 2: Antecedentes.

En este capítulo se presentan los antecedentes que permiten entender los conceptos básicos que se manejan para este trabajo. Así, se comienza con el concepto de palabras clave que sirve para categorizar un texto completo, después, se presenta la idea general del modelado de tópicos y cómo se relaciona con la minería de textos. Posteriormente, se habla sobre los métodos de modelado de tópicos, enfatizando en dos muy relevantes: *Latent Dirichlet Allocation[2]* y *Non-negative Matrix Factorization[3]*, para después presentar una forma de estudiar los tópicos en el tiempo y finalmente, hablar del sistema actual para las *Tesis UNAM*.

2.1. Palabras clave

El interés por conocer los temas que abordan las grandes colecciones de textos han empujado a los investigadores y desarrolladores a acuñar conceptos importantes para atacar esta problemática. El primer acercamiento que se tiene con este fin se remonta tiempo atrás, cuando los acervos y los investigadores empezaron a añadir el campo de palabras clave en sus fichas bibliográficas [4]. Con esto, si haces una revisión de las palabras clave de un conjunto de documentos puedes tener una idea de que temas tocan principalmente. Dicha actividad es muy popular y reconocida por la mayoría de las personas que ha hecho alguna investigación bibliográfica. En este caso, la idea es agregar a la ficha un campo adicional con algunas palabras elegidas por el creador de la misma, las cuales a su parecer, representan los temas de interés que toca el texto y con ese conjunto más pequeño de palabras se facilita el acceso a algunos temas del contenido. En la actualidad, la anterior estrategia ha sido modificada en cierta medida puesto que en varios ámbitos, como lo es el mundo de los artículos científicos, al creador del artículo se le pide que agregue alguna cantidad de palabras clave, normalmente tres o más.

En sus orígenes, el método antes mencionado solía ser común para uso personal pero no para ser proporcionada como información pública debido a que es claramente subjetiva bajo la percepción del creador de la ficha. Un ejemplo típico de ella podría ser el siguiente:



Ficha Bibliográfica	
Autor	: Van Eemeren, Frans; Grootendorst, Rob.
Título	: Argumentación, Comunicación y Falacias. Una perspectiva pragmadialéctica.
Año	: 2002.
Editorial	: Universidad Católica de Chile.
Ciudad	: Santiago.
País	: Chile.
Resumen del contenido	: Luego de contextualizar el enfoque pragmadialéctico, los autores realizan una introducción a los puntos de vista y diferencias de opinión, para luego explicar su visión de la argumentación como un acto de habla complejo, con especial énfasis en el análisis y evaluación del discurso argumentativo. Desde el capítulo IX y hasta el capítulo XIX, los autores se dedican al estudio de falacias.
Relevancia para nuestra investigación	
Tema de nuestra investigación	: Modelo argumentativo de Stephen Toulmin como criterio de evaluación de un ensayo.
Relevancia de esta obra para nuestra investigación	: Esta obra es relevante en gran medida, ya que realiza un estudio detallado desde el acto de habla hasta la estructura argumentativa, encontrándose ahí la base de nuestro estudio.
Al menos 5 palabras clave (hashtag)	: Argumentación, Comunicación, Falacias, Pragmadialéctica, Discurso.

Figura 2.1: Ejemplo de ficha bibliográfica con campo de palabras clave [1]

1 *Obtenida de https://www.academia.edu/34738336/Ejemplo_de_ficha_bibliogr%C3%A1fica._Razonamiento_cient%C3%ADfico (visitado 1 agosto 2020)*

Posteriormente con el avance tecnológico del siglo XXI, la idea se popularizó alrededor del mundo y empezó a ser de ayuda al incorporarse ese campo a los catálogos digitales y las búsquedas bibliográficas computarizadas. De tal forma que una biblioteca o acerbo en general, que tuviera implementado un sistema de base de datos para el manejo y consulta de sus ejemplares, y que además, se le hubiera incluido el campo de palabras clave, podría proporcionarle al usuario una búsqueda en este campo y obtener resultados que coincidieran exactamente con su búsqueda. Aquí se notan algunos inconvenientes. Si bien es de ayuda el tener marcadores sobre los temas (palabras clave), la forma de obtenerlos es difícil porque esencialmente las hace una persona usando su particular criterio. Al ser de esta manera, es muy posible que deje fuera ideas o términos que, desde su punto de vista, no le parecieron relevantes pero que para otros fines sí podrían serlo.

Otra desventaja que se aprecia es la coincidencia exacta con los términos de búsqueda y todo lo que eso puede implicar; por ejemplo: una palabra clave asignada a un documento podría ser “micro-chip”. Si alguien buscara “microchip” o “electrónica” ese documento no aparecería.

En la actualidad, después de que estalla la era del Internet y con esto los grandes buscadores como Google, Yahoo, DuckDuckGo, etcétera, esta idea ─aunque modificada y adecuada─ es la que se utiliza para realizar búsquedas entre las páginas web y resulta de suma importancia en la mercadotecnia digital usando los llamados *search engine optimization* (SEO). La modificación que ocupan es costosa ya que básicamente se consideran todas las palabras como palabras clave.

Una aplicación que usa exactamente el mismo concepto que se ve en la Figura 2.1 es *twitter* cuando emplea los *hashtag (#).* En este caso se refleja el problema que se comentaba, no hay un control universal en esto y cualquier *“tweet”* al que se agregue uno de estos indicadores será rastreado aunque no tenga una relación fuerte.

Hay que resaltar que, en principio, si no se sabe de qué habla el documento es difícil obtener palabras clave para él y esta complicación se eleva considerablemente cuando la cantidad de documentos es muy grande.

2.2. Minería de textos

La minería de textos es una variante particular de la minería de datos donde el objeto de minado son precisamente textos. Se le llama minería de textos al grupo de métodos y modelos que nos sirven para obtener información adicional de un conjunto de textos que no está explícitamente escrita en ninguno de ellos pero aparece al analizarlos todos juntos y en grandes cantidades. Algunos de los objetivos que se cumplen con estas técnicas son:

- Clasificación documental.
- Extracción de información grupal.
- Análisis de metatextos.
- Elaboración de resúmenes.

2.3. Modelado de tópicos

El modelado de tópicos en general se puede describir como una familia de métodos con los cuales se pueden encontrar distintos grupos de palabras interrelacionadas dentro de una colección de textos. A cada uno de estos grupos se le llama tópico. Dicho de otra forma, la aplicación del método entrega como resultado varios grupos de palabras que están de alguna forma conectadas, de tal manera que cada conjunto forma una especie de campo semántico el cual podría ser posteriormente etiquetado por un humano con su núcleo semántico. A continuación se presenta un ejemplo esquemático.

Ejemplo: Imagínese una colección de 5 *tweets*. Cada uno es considerado un documento.

1. CoDi, inicia adiós al efectivo.
2. Se suscitó un accidente dentro de la Feria de Chapultepec donde se desprendió uno de los carros mecánicos de un juego.
3. Ahora sí, autoridades revisaran ferias en CDMX para evitar futuros accidentes.
4. En China hasta el carro de comida callejera se paga con un CoDi.
5. No más efectivo ni tarjetas, de ahora en adelante paga con tu teléfono.

El resultado esquemático de usar un modelado de tópicos en estos documentos se muestra como la siguiente Tabla 2.1:

Palabras seleccionadas por el modelado de tópicos	**Nombre asignado por el usuario**
CoDi, efectivo, paga, carro	Sistema de cobro digital
Feria, accidente, carro	Accidente en la feria de Chapultepec

Tabla 2.1: Resultados ilustrativos de modelado de tópicos en tweets.

En este ejemplo los 5 *tweets* hablarían de sólo 2 temas distintos y el método entregaría los dos grupos de palabras para que posteriormente el usuario asigne un nombre que lo etiquete.

Hay que recordar que el modelado de tópicos se hace con colecciones grandes de documentos, dependiendo del método es la cantidad real necesaria, usualmente se ocupan como mínimo 2,000 documentos.

Existen varios métodos distintos para hacer modelado de tópicos y la interrelación entre palabras antes mencionada se percibe diferente de uno a otro. En esta línea de pensamiento

tenemos una forma de definir tópico y es: “un patrón repetitivo de co-ocurrencia de términos dentro de una colección”[5].

Actualmente existen varios métodos para el modelado de tópicos y cada uno puede tener distintas variantes dando lugar a una multiplicidad de algoritmos, cada uno con sus pros y contras. Para fines de este trabajo se considerarán como los más relevantes los siguientes dos:(1) *No negative matrix factorization* [3] *(NMF)* y (2) *Latent Dirichlet allocation* [2] (LDA). Siendo este último el de mayor relevancia en los últimos años debido a sus buenos resultados. Existe un tercer método que vale la pena mencionar por su relevancia historica pero ha quedado superado por los anteriores y es *Probabilistic latent semantic indexin* [6] (pLSI) por lo que no se ahondará en el durante este trabajo.

Otra cosa a resaltar es que cualquiera de estos métodos en sí mismos identifica las palabras relacionadas sin conocimiento previo de los textos, genera los tópicos y los deja listos para nombrarse si es necesario. Es notable el beneficio que se adquiere al facilitar la organización de las colecciones bajo estos métodos e incluso al aplicarlos se pueden descubrir tópicos ocultos a simple vista. El hecho de encontrar información que no se tenía antes es relevante y es justamente el objetivo de lo que llamamos minería de datos; en este caso específicamente se le llama minería de textos ya que los datos que se usan están en forma de texto.

2.4. Métodos de modelado de tópicos

En esta sección se describe, a grandes rasgos, el camino que tomó la investigación mundial para llegar a los métodos de mayor relevancia para el modelado de tópicos y posteriormente se hablará en términos globales sobre dos de ellos, *Latent Dirichlet Allocation* (LDA) y *Matrix Factorization* (MF).

Para llegar al desarrollo del modelo LDA se usaron los avances obtenidos por G. Salton y M. McGill publicados en su libro *Introduction to Modern Information Retrieval* (IR) [7] así como los posteriores de Baeza-Yates y Ribeiro-Neto en su artículo *Modern Information Retrieval* [8]. Sin embargo el considerado inmediato predecesor de este modelo es el *Latent Semantic Indexing* (LSI) [9] propuesto por Scott Deerwester et al. en 1990.

Revisando los avances obtenidos por los investigadores de IR [8] encontramos que la propuesta general de esta metodología es reducir cada documento de la colección a un arreglo de números reales, los cuales representan proporciones de conteos. Es aquí donde surge el popular esquema *tf-idf* del inglés *Term frequency – Inverse document frequency* que, en su forma más sencilla, es seleccionar un conjunto de "términos" o "palabras", contar las veces que aparece cada uno en la colección y multiplicarlo por el logaritmo del cociente del número de documentos y la cantidad de documentos en los cuales aparece ese término como se ve en la ecuación (3), donde se pueden usar :

- "Frecuencias" booleanas: *tf(t,d)* = 1 si t ocurre en *d*, y 0 si no;
- Frecuencia escalada logarítmicamente: *tf(t,d)* = 1 + log *f(t,d)* (y 0 si *f(t,d)*=0);
- Frecuencia normalizada, para evitar una predisposición hacia los documentos largos como se describe en (1).

$$tf(t,d) = \frac{f(t,d)}{max(f(t,d) : t \in d)} \tag{1}$$

$$idf(t,D) = \log \frac{|D|}{|d \in D : t \in d|} \tag{2}$$

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) \tag{3}$$

En la ecuación (2) $|D|$ es la cantidad de documentos y el denominador es el número de documentos donde aparece el término *t*.

Una consecuencia de este planteamiento es: mientras mayor sea el *idf* para un término en particular, más importante será en el documento. Esto es así porque hay palabras más comunes que otras en los textos y cuando alguna aparece con mucha frecuencia resulta menos importante. En cierta forma, éste es un método matemático para obtener palabras relevantes lo cual se parece a la estructura de palabras clave que se mencionó en la sección 2.1. Actualmente se utiliza *tf-idf* en los buscadores de Internet y sus algoritmos conocidos como SEO.

Después de aplicar el método a una colección lo que se obtiene es una matriz X_{NxM} donde N es el número de documentos y M la cantidad de términos elegidos como vocabulario. En sus entradas contiene el valor *tf-idf* para cada término en cada documento. Una ventaja remarcable de la aplicación es: dado que el conjunto de términos es fijo y preseleccionado, entonces se reduce la dimensión de un tamaño arbitrario de términos que podrían estar contenidos en cada documento a un arreglo de tamaño fijo de números reales.

La ventaja antes mencionada es importante pero no es suficiente dado que nos proporciona muy poca información de las relaciones de los términos entre los documentos, o incluso dentro de uno solo; además la reducción en la dimensión no siempre es suficiente.

La siguiente propuesta nace con el método conocido como Latent Semantic Indexing [9]. Lo que plantea LSI es hacer una descomposición en valores singulares de la matriz *X* para identificar un subespacio lineal de *tf-idf* tal que capture la mayor parte de la varianza en la colección. Esta estrategia sí puede otorgar una reducción significativa para colecciones grandes de textos.

La anterior técnica es la primera que puede capturar nociones básicas del lenguaje como sinónimos y polisemias[2]. Por ello, en aquel momento, se motivó la comunidad

2 Según https://www.significados.com/polisemia/: La polisemia es un término que se emplea para denominar la diversidad de acepciones que contienen determinadas palabras o signos lingüísticos

investigadora planteándose nuevas preguntas sobre modelos probabilísticos similares que pudieran identificar patrones dentro de los textos.

En 1999, Hoffman propuso el siguiente paso importante cuando publicó su probabilistic Latenen Semantic Indexing [6] (pLSI). Este modelo, al igual que LSI, utiliza el importante concepto de utilizar cada documento como una "bolsa de palabras", lo cual quiere decir que el orden en el que aparezcan las palabras resulta irrelevante. También trata a cada palabra como una muestra de un modelo de mezclas probabilístico, donde los componentes de la mezcla son variables multinomiales aleatorias que pueden ser vistas como representaciones de un "tópico". Siendo así, cada palabra es generada por un único tópico. Cada documento es representado como una combinación de las partes que componen a la mezcla y así se pueden reducir a una distribución de probabilidad de un conjunto dado de tópicos.

Los avances proporcionados por pLSI son importantes pero el modelo resulta incompleto ya que no provee un modelado probabilístico a nivel de documento. Esto es que cada documento está representado como una lista de números (los valores de las proporciones de tópicos dentro de la mezcla) y no existe un modelo para generarlos. Uno de los problemas más grandes que se derivan de pLSI es que el número de parámetros en el modelo crece linealmente con el tamaño de la colección, lo cual, a tamaños muy grandes resulta inmanejable.

2.4.1. Latent Dirichlet Allocation

En el 2003 David M. Blei, Andrew Y. Ng y Michael I. Jordan publicaron un artículo llamado *"Latent Dirichlet Allocation"* [2] (LDA) donde propusieron un modelo probabilístico para el modelado de tópicos de grandes colecciones de textos. Se dice que es un modelo de tres niveles de jerarquización Bayesiana, en donde cada elemento de una colección está modelado como una combinación finita de un conjunto de tópicos

subyacentes. Cada tópico, a su vez, es modelado como una combinación infinita de probabilidades.

Es importante resaltar que a diferencia de lo que se logra con pLSI, LDA sí es un modelo probabilístico generativo de una colección de textos. La gran idea detrás es que los documentos son representados como una distribución probabilística de tópicos latentes y donde cada tópico está identificado con una distribución probabilística de las palabras.

Esbozando el método se dice que LDA usa dos valores de probabilidades: el primero es la probabilidad de una palabra dado el tópico, *P(word|topic)* y el segundo, es la probabilidad de un tópico dado un documento, *P(topic|doc)*. Estos valores se calculan a partir de una asignación aleatoria inicial y mediante un proceso iterativo se van calculando múltiples veces hasta que el algoritmo converge.

Para detallar, primero se tiene que definir el proceso de creación de un documento matemáticamente. La notación y terminología necesaria se muestra en la Tabla 2.2:

V	*Tamaño del vocabulario*
w	*Representa una palabra. Es la unidad básica de "discrete data". Ésta se escribe como un arreglo de tamaño V {1,…,V} y cada una está asociada a un vector unitario con una entrada igual a uno y todas las demás iguales a cero.*
N	*Cantidad de palabras en cada documento*
w	*Representa un documento y es una secuencia de N palabras denotadas por* $\mathbf{w}=(w_1,w_2,\ldots,w_N)$*, esto es un arreglo de "w's"*
k	*La cantidad de tópicos que hay en la colección.*
M	*Cantidad de documentos.*
D	*La colección completa. Un conjunto de M documentos.* $D=\{\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_M\}$.
z	*Un tópico de una colección de k tópicos. Un tópico es una distribución de palabras. Por ejemplo: Animal = (0.3 Cats, 0.4 Dogs, 0 AI, 0.2 Loyal, 0.1 Evil)*

Tabla 2.2: Notación y terminología para LDA.

En la Figura 2.2 tenemos un único valor de α *(organizador del campo theta)* la cual define como será la distribución de tópicos para los documentos, a la que llamamos θ. Tenemos *M* documentos y una distribución θ para cada uno de ellos. Ahora, cada documento tiene *N* palabras y cada palabra es generada por un tópico. Se generan *N* tópicos para ser llenados con palabras. Estas *N* palabras son marcadores de posición.

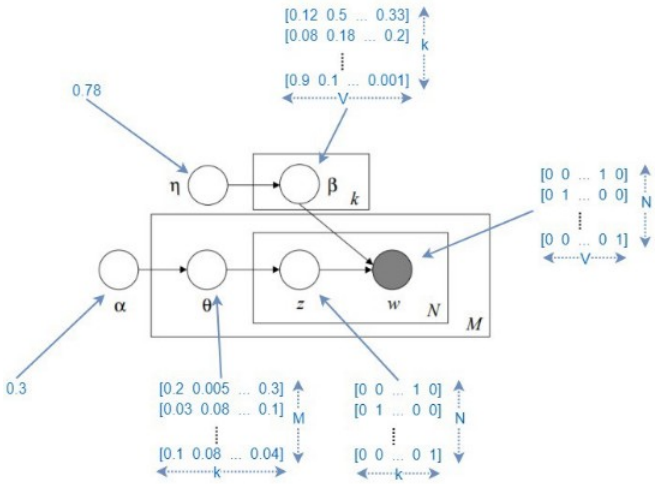


Figura 2.2: Diagrama de generación de documentos matemáticamente para LDA.[3]

En la parte superior se observa algo similar a partir de η, β tiene una distribución, la cual resulta conveniente sea una distribución de Dirichlet[10]. De acuerdo con ella, β genera *k* palabras para cada tópico. Por último se llena una palabra para cada uno de los *N* marcadores de posición condicionada que representa al tópico para finalmente obtener el documento.

En la Figura 2.2 θ y β parecen escalares pero en realidad son matrices. θ es una matriz aleatoria en la que $\theta(i,j)$ representa la probabilidad de que el *i*-ésimo documento contenga palabras del *j*-ésimo tópico. En un principio, se pretende que las palabras generadas pertenezcan principalmente a un solo tópico; es por ello que θ inicial se plantea como una distribución de Dirichlet. Esta distribución tiene la característica que se acumula más en las esquinas y menos en el centro, alejando las palabras entre tópicos. Esto simula lo que pasa en los documentos reales adecuadamente. De forma similar, $\beta(i,j)$ representa la probabilidad de que el *i*-ésimo tópico contenga la *j*-ésima palabra. También se usa una

3. *Obtenida de https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158 (visitado 1 de agosto 2020)*

distribución de Dirichlet para ella. Ambas matrices contienen la información de la idea inicial de *P(word|topic)* y *P(topic|doc)*.Dicho esto, se puede ejemplificar la generación de documentos bajo el paradigma LDA como sigue: Por simplicidad, supóngase que se genera un solo documento con 5 palabras pero es claro que el proceso se puede generalizar a *M* documentos con *N* palabras en cada uno.

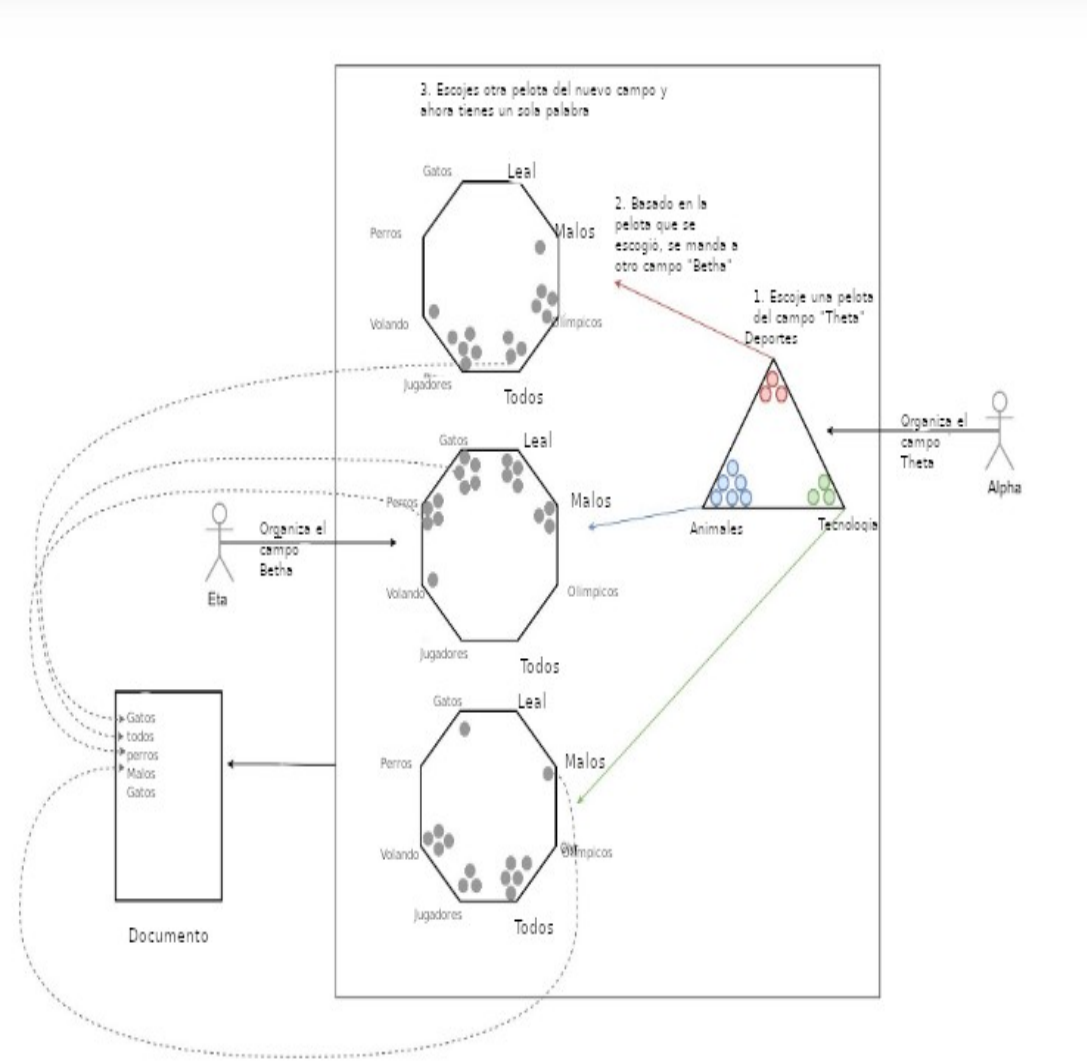


Figura 2.3: Esquema del generado de un documento mediante LDA.[4]

4. *Traducido de: ttps://htowardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158 (visitado 3 de agosto 2020)*

A continuación se presenta un ejemplo para ilustrar la idea de latencia de tópicos y por qué es importante.

Imagínese un primer escenario en el que el objetivo es encontrar a que categoría pertenece cada documento en un conjunto de 1,000 documentos y que se utilizó un vocabulario de 1,000 palabras para escribirlos, también supóngase que cada documento tiene 500 palabras en promedio. Ahora, lo que se puede hacer es conectar cada documento con cada palabra dentro de él y así observar una relación como se muestra en la Figura 2.4.

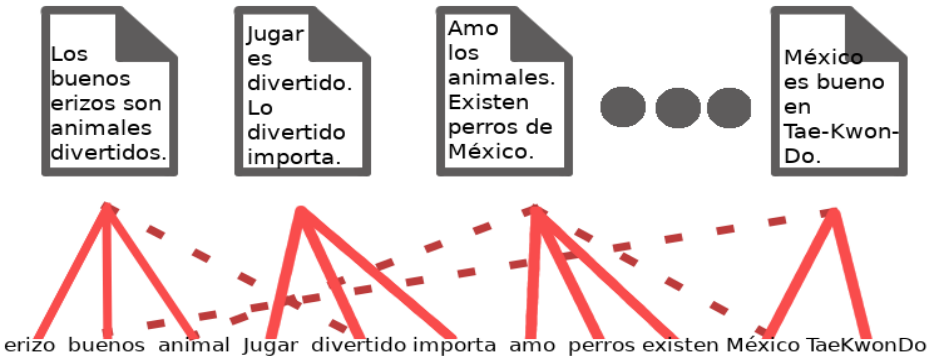


Figura 2.4: Representación de las conexiones de documento con vocabulario.[5]

De esta forma se puede ver que algunos documentos se conectan con los mismos subconjuntos de palabras y así se puede inferir que los que comparten subconjuntos están hablando del mismo tema; sin embargo, hacer todas las conexiones demanda bastante trabajo, *500*1,000=500,000* conexiones aproximadamente.

Para disminuir las conexiones se introduce la siguiente idea importante: una capa de latencia de los tópicos. Esto se hace bajo la premisa que se sabe o se estima con algún otro método alejado a LDA (puede ser simple arbitrariedad del usuario), cuantos tópicos hay en

5 Inspirado en: https://htowardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158 (visitado 3 de agosto 2020)

toda la colección. En este ejemplo, supongamos que son 10 tópicos y lo único que se ve son palabras y documentos. Lo siguiente que se hace es conectar las palabras con los tópicos dependiendo de que tan bien encajen (hay que recordar que se dijo que los tópicos son una distribución de palabras) y después conectar cada tópico con los documentos que tratan sobre éste.

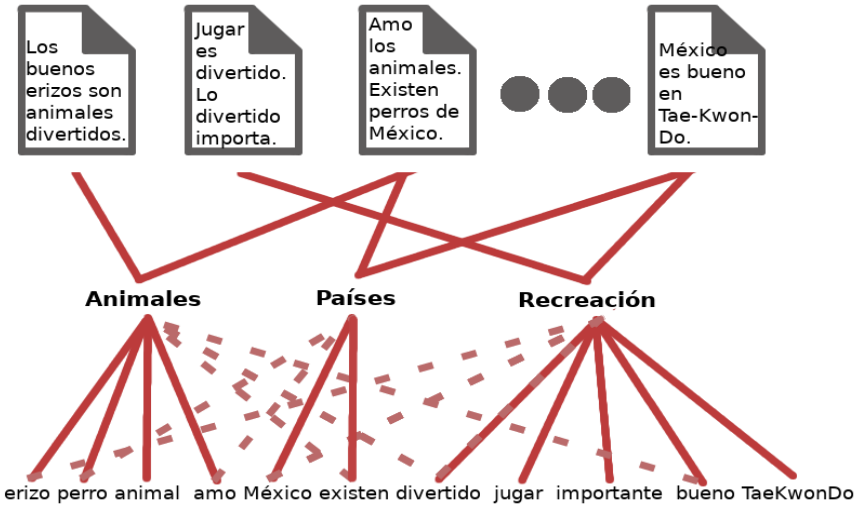


Figura 2.5: Representación de las conexiones utilizando tópicos latentes.[5]

Hay que resaltar que en la Figura 2.5 los tópicos “Animales, Países y Recreación” están adaptados para el ejemplo; en realidad se verían como palabras con una probabilidad asociada. El tópico Animales se vería algo semejante a esto: erizo(0.5), perro(0.5), animal(1.0), amo(0.5), existen(0.5), divertido(0.5)

Finalmente, supongamos que cada documento trata 5 tópicos diferentes y que cada tópico está relacionado con 500 palabras, entonces se necesitan 1,000x5 conexiones para unir los documentos a los tópicos y 5x500 conexiones para unir las palabras con los tópicos. Esto nos da un total de 2,500 conexiones únicamente.

Teniendo en cuenta todo lo anterior podemos plantear el problema en forma de una probabilidad de la siguiente manera:

$$P(\theta_{1:M}, z_{1:M}, \beta_{1:k} | D; \alpha_{1:M}, \eta_{1:k}) \quad (4)$$

En este trabajo no se ahondará en los detalles del algoritmo y la programación.[6]

2.5. Non-negative Matrix Factorization

Non-negative Matrix Factorization (NMF) es un nombre general que se le da a un grupo de métodos que funcionan bajo el mismo principio que, como su nombre lo indica, factoriza una matriz *A* en otras dos *W* y *H*. La matriz inicial *A* contiene la información de las palabras en los documentos. Al hacer la factorización, una de las matrices que se obtienen contiene la información de los tópicos de la colección.

El planteamiento que hacen Jaegul Choo, Changhyun Lee, Chandan K. Reddy y Haesun Park en su artículo "UTOPIAN: user-driven topic modeling based on interactive nonnegative matrix factorization" (2013) [3] es como sigue:

m	*Numero de palabras clave*
n	*Número de documentos*
k	*Número de tópicos*
$X \in \mathbb{R}^{m \times n}$	*Matriz término-documento*
$W \in \mathbb{R}^{m \times k}$	*Matriz término-tópico*
$H \in \mathbb{R}^{k \times n}$	*Matriz tópico-documento*
$w_l \in \mathbb{R}^{m \times l}$	*Representación palabras clave de el l-ésimo tópico*
$h_i \in \mathbb{R}^{k \times l}$	*Representación en tópicos de el i-ésimo documento*

6 Algunos links donde se habla del algoritmo: https://medium.com/@lettier/how-does-lda-work-ill-explain-using-emoji-108abf40fa7d
https://towardsdatascience.com/lda-topic-modeling-an-explanation-e184c90aadcd
https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158

$V \in \mathbb{R}^{m x k}$	*Referencia término-tópico de la matriz W*
$G \in \mathbb{R}^{k x n}$	*Referencia tópico-documento de la matriz H*
$v_l \in \mathbb{R}^{m x l}$	*Vector referencia para* w_l
$g_i \in \mathbb{R}^{k x l}$	*Vector referencia para* h_i
$M_W \in \mathbb{R}^{k x k}$	*Una matriz pesada para las columnas de W*
$M_H \in \mathbb{R}^{n x n}$	*Una matriz pesada para las columnas de H*
$M_W^{(l)}$	*Un valor pesado para* w_l
$M_H^{(i)}$	*Un valor pesado para* h_i

Tabla 2.3: Notación para NMF.

Dada una matriz no negativa $X \in \mathbb{R}_{m_x n}$, y un entero $k \ll min(m, n)$ se dice que NMF obtiene una aproximación de menor rango tal que:

$$X \approx WH \tag{5}$$

donde $W \in \mathbb{R}_{n_x k}$ y $H \in \mathbb{R}_{k_x m}$ son factores no negativos de *X*. La manera más común de plantear el problema es en términos de la Norma de Frobenius[7] tal que:

$$\min_{W,H \geq 0} \|X - WH\|_F^2. \tag{6}$$

donde ≥ se aplica a todos los elementos de la matriz dada en el elemento del lado izquierdo.

La forma en que se calculan *W* y *H* es mediante una optimización sobre una función objetivo, en la cual se actualizan los valores de *W* y *H* de forma iterativa hasta que converge[11].

$$\frac{1}{2}\|X - WH\|_F^2 = \sum_{i=0}^{n} \sum_{j=1}^{m} (X_{ij} - (WH)_{ij}) \tag{7}$$

7 https://es.wikipedia.org/wiki/Norma_matricial#Norma_de_Frobenius
http://esfm.egormaximenko.com/linalg/Frobenius_norm_es.pdf

La ecuación (7) representa la función objetivo.

En esta función se mide el error de reconstrucción entre *X* y el producto de *W* por *H*. Las reglas se pueden derivar y se obtiene:

$$W_{ic} \frac{(\boldsymbol{X}\, H)_{ic}}{(\boldsymbol{W}\, \boldsymbol{H}\, H)_{ic}} \rightarrow W_{ic} \tag{8}$$

$$H_{cj} \frac{(W\, \boldsymbol{X})_{cj}}{(W\, \boldsymbol{W}\, \boldsymbol{H})_{cj}} \rightarrow H_{cj} \tag{9}$$

Con c indicando un valor fijo para cada asignación.

En el contexto del modelado de tópicos la *i*-ésima columna de *X*, $x_i \in R_{m_x 1}$, corresponde a la bolsa de palabras que representa al documento *i* con respecto a *m* palabras marcadoras. Estos valores pueden llevar algún pre-procesamiento como *"lista invertida de documento pesado por frecuencias" y "normalización por columnas mediante norma-L2"*. El escalar *k* representa el número de tópicos. La *l*-ésima columna de *W*, escrita como w_l, representa el *l*-ésimo tópico como una combinación ponderada de *m* palabras. Cuando se tiene un valor elevado, se dice que existe una fuerte correlación ente la palabra y el tópico. Así mismo, se dice que la *i*-ésima columna de *H* escrita como h_i, representa al *i*-ésimo documento como una combinación ponderada de *k* tópicos.

Con lo dicho anteriormente, se pueden comparar los resultados obtenidos a través de este método y lo que entrega LDA o pLSI y se puede afirmar que los resultados son equivalentes. Por un lado se obtienen conjuntos de palabras que representan los tópicos descritos por los w_l y por otro lado se obtiene una serie de vectores que relacionan los documentos con los tópicos que le corresponden descritos por los h_i.

Un ejemplo de cómo se ve el método se ilustra en la Figura 2.6:

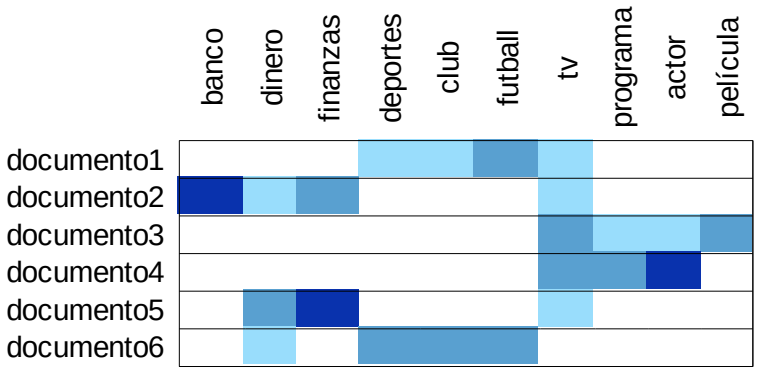


Figura 2.6: Representación de la matriz X para NMF. Mientras más obscuro es el color mayor es la relación.

Un ejemplo después de aplicar el método es el siguiente:

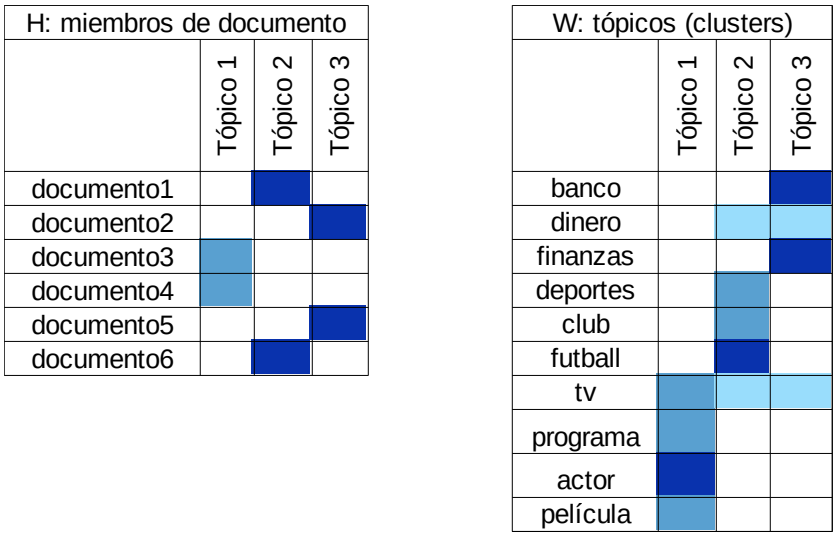


Figura 2.7: Ejemplo de matrices factores W y H. Mientras más obscuro es el color mayor es la relación.

NMF es un método popular actualmente y ha llamado la atención desde que Simon Funk propuso el algoritmo original en su blog del 2006 [12]. A partir de ese momento han surgido varias mejoras y modificaciones útiles para objetivos en particular de los cuales el más conocido podría ser el algoritmo de sugerencias con base en lo visto anteriormente para la plataforma Netflix.

En este trabajo se realiza el modelado de tópicos mediante otro método llamado *Sampled Weighted MinHashing* el cual se detalla en el Capitulo 3.

2.6. Tópicos en el tiempo

Una vez que se obtienen los tópicos de una colección de textos el siguiente análisis que se puede hacer es ver cómo es que estos tópicos se relacionan con otros en el tiempo. Nótese que los algoritmos descritos en las secciones anteriores utilizan toda la colección sin alguna separación temporal y a partir de eso se calculan las probabilidades o los parámetros pertinentes para identificar los tópicos.

Es relevante ubicar los tópicos en una línea temporal ya que con esto se pueden diferenciar tópicos efímeros o de “moda” de otros que perduran en el tiempo.

Este comportamiento es interesante dado que refleja tendencias o preferencias en los documentos que pueden quedar perdidas al analizar toda la colección por igual. Incluso, teniendo un buen modelo, se puede predecir el comportamiento a futuro de los tópicos, si estos van a desaparecer o se juntarán con otro, e incluso si uno nuevo va a ser relevante en el futuro.

Existen varios modelos que llevan a cabo esta labor, uno de ellos fue planteado por Xuerui Wang, Andrew McCallum y lo llamaron Topics Over Time (TOT)[13] en su artículo del 2006. En él dicen que el modelado de tópicos con TOT no solo se ve influenciado por la co-ocurrencia de palabras sino que también por la información temporal que se tenga de los documentos. A diferencia de otros modelos anteriores, éste es capaz de obtener marcas de tiempo absolutas o globales, lo cual permite ver dependencias en el tiempo a largo plazo que se pueden usar para predecir valores absolutos de tiempo para un documento dado sin clasificación temporal, y también predecir la distribución de los tópicos para un tiempo dado.

Otra de las virtudes que presenta TOT es que no maneja el tiempo de forma discreta. Otros métodos utilizan rebanadas para esta variable y normalmente resulta que la elección de su tamaño es muy grande para algunas regiones y muy pequeña para otras. La forma en que se evita es asociando cada tópico con una distribución continua en el tiempo. Se pueden usar muchas distribuciones pero las más usuales son basadas en Gaussianas y distribuciones Beta.

TOT es un modelo generativo de marcas temporales y de las palabras en los documentos marcados temporalmente. Existen dos formas de describir el proceso generativo. La primera corresponde al proceso usando muestreo de Gibbs[8] para la estimación de parámetros y es el siguiente:

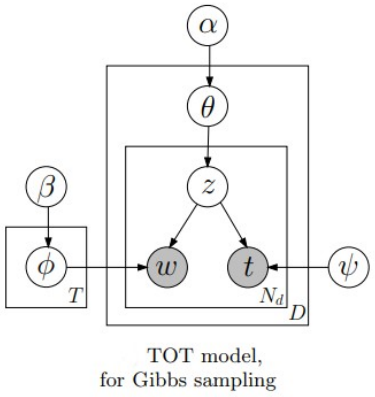


Figura 2.8: modelado TOT [9]

La notación que se usa es la siguiente:

α	*Vector parametrizante de la distribución* θ
β	*Vector parametrizante de* Φ

8 http://www.dpye.iimas.unam.mx/eduardo/MCB/node27.html
9 Imagen obtenida de [13] directamente.

T	*Cantidad de tópicos*
D	*Cantidad de documentos*
V	*Cantidad de palabras únicas (vocabulario)*
N_d	*Cantidad de palabras en el documento d*
θ_d	*Distribución multinomial de tópicos específicos para el documento d*
Φ_z	*Distribución multinomial de palabras especificas para el tópico z*
ψ_z	*Distribución beta para el tiempo de el tópico z*
z_{di}	*El tópico asociado con la i-ésima palabra en el documento d*
w_{di}	*La i-ésima palabra en el documento d*
t_{di}	*Marca de tiempo asociada a la i-ésima palabra en el documento d*

Tabla 2.4: Notación para TOT.

En la Figura 2.8 se ve que primero se toma T cantidad de Φ multinomiales de una Dirichlet a partir de β, una para cada tópico z. Después, para cada documento d, se toma una multinomial θ_d de una Dirichlet a partir de α; luego para cada palabra w_{di} en el documento d se hacen tres cosas: (1) se toma un tópico z_{di} de la multinomial θ_d, (2) se toma una palabra w_{di} de la multinomial $\phi_{z_{di}}$ y, (3) se toma una marca temporal t_{di} de la $\beta(\psi_{z_{di}})$.

Obsérvese que en el proceso anterior se genera una marca temporal para cada palabra, esto es, todas las marcas temporales de las palabras en un documento se verán iguales que la marca temporal del documento mismo.

Recalcando, el proceso anterior entrega una distribución de tópicos que depende de dos partes por igual, el tiempo y el texto. La parametrización que se usa es:

$$
\begin{aligned}
\theta_d|\alpha &\sim \text{Dirichlet}(\alpha) \\
\phi_z|\beta &\sim \text{Dirichlet}(\beta) \\
z_{di}|\theta_d &\sim \text{Multinomial}(\theta_d) \\
w_{di}|\phi_{z_{di}} &\sim \text{Multinomial}(\phi_{z_{di}}) \\
t_{di}|\psi_{z_{di}} &\sim \text{Beta}(\psi_{z_{di}}).
\end{aligned}
$$

Figura 2.9: Parametrización de TOT .

La inferencia no se puede hacer de manera exacta en este modelo. Es por eso que se usa el muestreo de Gibbs para aproximar la solución. Wang y otros[13] en su artículo de 2006 detallan el algoritmo para realizar el muestreo.

Después de todo, un documento se modela como una mezcla de tópicos y usualmente solo existe una marca temporal asociada a cada documento. Por otro lado, el proceso generativo explicado anteriormente describe datos en los cuales hay una marca de tiempo asociada a cada palabra.

El modelo tiene una deficiencia. Al ajustarlo con datos típicos, sucede que la marca de tiempo de cada documento se copia a todas las palabras en ese documento. Sin embargo, si se utiliza como un modelo generativo, este proceso generaría marcas de tiempo distintas para las palabras contenidas dentro del mismo documento.

La otra forma de describir el proceso generativo es la descrita en el siguiente diagrama:

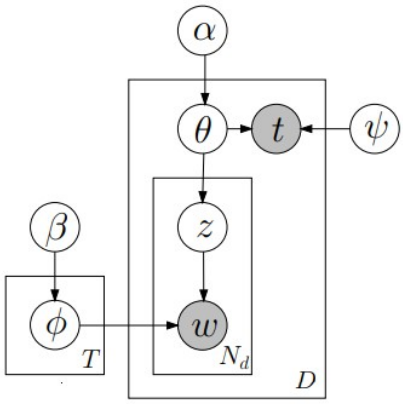


Figura 2.10: Vista alternativa de TOT. [9]

En esta alternativa se plantea que solo una marca temporal sea asociada con cada documento, generada por rechazo o muestreo de importancia a partir de una mezcla de distribuciones β por tópico en el tiempo con los pesos de mezclas θ_d por documento sobre los tópicos. Con este modelo se puede predecir la marca temporal dadas las palabras en el documento.

Otra cosa que es interesante es obtener una distribución de los tópicos en función de la marca temporal. Esto permite ver los patrones de ocurrencia de tópicos a través del tiempo.

$$E(\theta_{zi}|t) = P(z_i|t) \propto P(t|z_i)\,P(z_i) \qquad (10)$$

Por la regla de Bayes, donde *P(z)* puede ser estimada con los datos o suponerse uniforme.

Finalmente, el método con el cual se elaborará la correlación final de la evolución de los tópicos en el tiempo es mediante uno más directo que se detalla en la sección 3.7.

2.7. Tesis UNAM

En este trabajo se utiliza como colección de documentos un subconjunto de la base de datos conocida como TESIUNAM.

TESIUNAM[14] es el sistema que contiene el catálogo en el cual se visualizan las tesis que por reglamento entregan a Biblioteca Central los sustentantes que obtuvieron un grado académico en la UNAM ya sea licenciatura, maestría o doctorado, así como las tesis de licenciatura de escuelas incorporadas a la UNAM. Es aquí donde se ha ido concentrando la colección de tesis más grande de México.

La implementación de TESIUNAM se remonta a partir del año 1985 como lo comenta Juan Voutssas[15] en las memorias de la XIX jornada de biblioteconomía. En un principio sólo manejaba las fichas bibliográficas de 60,000 tesis y únicamente dentro de la red de bibliotecas de la UNAM. En años posteriores se agregaron los documentos digitales y finalmente la consulta de las mismas desde Internet.

A partir del año 2004 se hizo obligatoria la entrega en formato digital de origen a todos los sustentantes. En general, las tesis anteriores a ese año están digitalizadas mediante escaneo físico y un buen porcentaje tienen un procesamiento bajo un reconocimiento óptico de caracteres (OCR).

Actualmente el catálogo de tesis, que abarca trabajos desde 1900 a la fecha, cuenta con más de 450,000 registros de los cuales aproximadamente 150,000 están en formato electrónico[16].

Capítulo 3: Minado de tópicos: SWMH.

En este capítulo se revisará con detalle el algoritmo de minado de tópicos desde su base matemática hasta la implementación.

3.1. Conjuntos semejantes

Desde 1901 P. Jaccard [17] se plantea la pregunta: ¿cómo evaluar si un par de conjuntos son o no parecidos? En su estudio él postula un coeficiente de similitud entre dos conjuntos A_1 y A_2 como una medida entre cero y uno, siendo cero para conjuntos ajenos y uno para conjuntos idénticos. Hoy en día se le conoce como coeficiente de similitud de Jaccard y está definido como:

$$simi(A_1, A_2) = \frac{|A_1 \cap A_2|}{|A_1 \cup A_2|} \tag{11}$$

Una forma de organizar una familia de n conjuntos A_1, A_2,…,A_n puede ser encontrando aquellos pares cuya similitud de Jaccard sea mayor que cierto umbral a considerar, el cual dependerá de la situación en particular, por ejemplo, se pueden organizar en dos secciones, la primera es aquellos conjuntos A_i que tengan por lo menos otro A_j cuya similitud de Jaccard sea mayor que el umbral elegido, digamos 0.8 y considerarlos “conjuntos semejantes”. La otra sección quedará conformada simplemente con los conjuntos A_i que no tengan algún otro “conjunto semejante”. Con esta evaluación se pueden identificar dentro de una familia de conjuntos aquellos pares suficientemente similares de una forma matemáticamente sencilla; sin embargo, dependiendo de los tamaños de los conjuntos y del tamaño de la familia misma puede representar un problema ya que calcular la intersección de todos los conjuntos podría requerir demasiadas operaciones.

3.2. Funciones Hash

Una función hash (o simplemente un hash) es cualquier función que a un dato de cualquier longitud -denominado llave- le asigna un valor de longitud fija[18]. Se usan los valores que entrega la función para indexar una tabla de tamaño fijo llamada “tabla hash”. A esta acción se le llama hashing.

Algunos de los usos más comunes para las funciones hash y sus tablas hash asociadas es en el almacenamiento de datos y en aplicaciones de recuperación (bases de datos) ya que permite acceder a los datos rápidamente y en un tiempo prácticamente constante sin importar las características del arreglo. Otro beneficio es que el espacio de almacenamiento es solo un poco más grande que el total requerido por los datos mismos. Este tipo de funciones son muy poderosas computacionalmente hablando ya que evita el incremento no-lineal del tiempo de ordenamiento de listas y árboles estructurados, y el aumento exponencial de espacio necesario para almacenar datos con una forma de acceso directo mediante llaves de longitud variable .

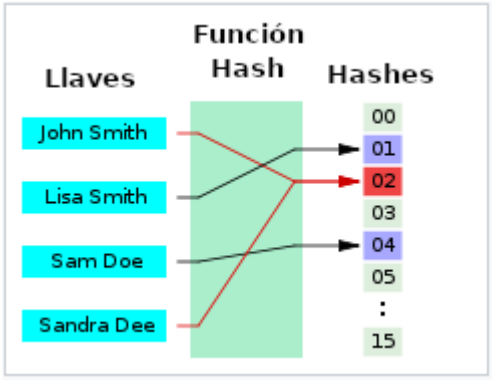


Figura 3.1: Diagrama de una función hash que mapea nombres a números del 0 al 15. Hay una colisión entre las llaves "John Smith" y "Sandra Dee".[10]

10 Traducido de https://en.wikipedia.org/wiki/Hash_function (visto en abril 2020)

Usualmente la forma de estas funciones está dada con una distribución de probabilidad, de tal suerte que toda llave es mapeada a un índice. La contraparte es que 2 llaves distintas podrían ser asignadas al mismo índice, cuando esto ocurre se le llama colisión. Es claro que la restricción para que no ocurra una colisión depende directamente del algoritmo que se implemente para el hash. Para algunas aplicaciones es más tolerable una colisión que para otras y con los algoritmos se puede ajustar que tan estricta resulta. Este trabajo se enfocará en los métodos que aumentan la probabilidad de colisión para llaves similares.

3.2.1. *Locality-sensitive hashing*

La expresión denominada *"locality-sesitive hashing" (LSH)* se refiere a una familia de algoritmos de mapeo mediante funciones hash que cumplen con la siguiente característica: Llaves de entrada similares tienen una alta probabilidad de ser mapeadas hacia la misma celda. Es decir, es altamente probable que dos entradas similares caigan en la misma celda.

3.2.2. Min-Hashing

El esquema denominado Min-wise independent permutation locality sensitive hashing o simplemente *Min-hashing* es un ejemplo de *LSH* muy importante para este trabajo ya que es la base del mismo. Para comprender este esquema basta con definir la función *h* de la siguiente manera:

Dados $U=\{a_1,a_2,...,a_v\}$ el conjunto universo y $A_i \subseteq U$ con *i={1,…,s}*. Sea π una permutación aleatoria de los elementos de *U* descrita por: $\pi(U)$, la cual induce un nuevo orden $U=\{a'_1,a'_2,...,a'_v\}$ y nos referimos a los elementos como $\pi(U)_k$.

Se define $h_\pi : \cup A_i \rightarrow U$, $A_i \rightarrow a'_{min|j|} : a'_j \in A_i$

Se puede construir h_π generando una permutación π de todos los elementos del conjunto universo, y después, asignando el primer elemento que aparece en la permutación y que pertenece al conjunto A_i como el valor de $h(A_i)$.

Por ejemplo:

Sean $U=\{a,b,c,d,e\}$ y $A_1=\{b,e,d\}$, $A_2=\{c,e\}$, $A_3=\{e\}$, $A_4=\{e,b,c\}$, $A_5=\{b,c\}$ y $A_6=\{d,e\}$

Ahora $\pi(U)$ induce un nuevo orden en U tal que $U=\{c,a,e,b,d\}$ Entonces

$h_\pi(A_1)=e$, $h_\pi(A_2)=c$, $h_\pi(A_3)=e$, $h_\pi(A_4)=c$, $h_\pi(A_5)=c$ y $h_\pi(A_6)=e$

3.2.3. Min-Hashing para búsqueda de pares de conjuntos similares

Una aplicación de las funciones min-hash es la búsqueda de pares de conjuntos similares. Este es un algoritmo probabilístico que sirve para encontrar dos conjuntos similares entre si dentro de una familia de conjuntos. La idea de esta aplicación es que dados varios conjuntos A_1, A_2,…, A_n se quiere encontrar cuales de ellos comparten muchos elementos en común. Dicho de otra forma, cuales de ellos son parecidos entre si.

Partiendo de la similitud de Jaccard descrita en (11), Chum et. al. [19] plantearon una expresión que se puede considerar una generalización de dicho coeficiente. Donde se considera la frecuencia con la que un elemento aparece en el conjunto y se asigna un término de ponderamiento. La llamaron *Weighted histogram intersection* y se presenta a continuación:

$$simi_h(A_1,A_2)=\frac{\sum_w d_w min(t_1^w,t_2^w)}{\sum_w d_w max(t_1^w,t_2^w)} \qquad (12)$$

Donde t_i^w se refiere a la coordenada *w* del vector t_i que contiene el numero de cuentas de una palabra X_w en el i-ésimo documento y d_w es un factor de peso para darle distinta importancia a las palabras. Obsérvese que dados dos conjuntos A_1, A_2 y una función *Min-hash* como se definió en la sección 3.2.2 se puede encontrar la probabilidad de que ambos conjuntos tengan el mismo valor de *h*. Esto se hace contando casos favorables, casos totales y simplemente haciendo el cociente:

$$P[h(A_1)=h(A_2)]=\frac{Casos\ Favorables}{Casos\ Totales}=\frac{|A_1\cap A_2|}{|A_1\cup A_2|} \tag{13}$$

Donde el cardinal de la intersección son los casos favorables, ya que son los elementos en común, y el cardinal de la unión son los casos totales. Nótese que este cociente es igual al coeficiente de similitud de Jaccard de tal forma que se puede entender como una manera probabilística de calcular el valor de Jaccard. Así se ve que los conjuntos similares tienen una alta probabilidad de tener el mismo valor Min-hash, al contrario que los conjuntos no similares que tienen una baja probabilidad de tener el mismo valor de *h*. Como casos especiales son los conjuntos disjuntos que $|A_1\cap A_2|=0$ y los conjuntos idénticos donde $A_1\cap A_2=A_1\cup A_2$ haciendo el cociente igual a uno.

Para evaluar la similitud entre dos conjuntos usando Min-hash primero, se toma una función *h* tal como se propone en la sección 3.2.2. y después, se calculan varios valores Min-hash para cada conjunto A_i a partir de distintas permutaciones aleatorias, estos múltiples cálculos son con el objeto de disminuir las fluctuaciones probabilísticas. Dicho de otra forma, primero se hace una permutación aleatoria para cada conjunto y luego, se calculan los respectivos valores Min-hash, se guardan y se repite todo el proceso muchas veces.

Usando lo anterior se construyen l tuplas g_1, g_2,...g_l de r entradas cada una con los valores Min-hash correspondientes de la siguiente manera:

$$\begin{aligned} g_1(A_1) &= (h_1(A_1), h_2(A_1), \ldots, h_r(A_1)) \\ g_2(A_1) &= (h_{r+1}(A_1), h_{r+2}(A_1), \ldots, h_{2r}(A_1)) \\ &\ldots \\ g_l(A_1) &= (h_{(l-1)r+1}(A_1), h_{(l-1)r+2}(A_1), \ldots, h_{lr}(A_1)) \end{aligned} \tag{14}$$

donde $h_j(A_1)$ es el j-ésimo valor Min-hash. De tal manera que se calculan $r\,x\,l$ permutaciones de los elementos de $\cup A_i$ y cada una de éstas induce un valor Min-hash. Así mismo, aplicando las g_k con $k=1,2,\ldots,l$ para todos los conjuntos A_i con $i=1,2,\ldots,s$ se construyen l diferentes tablas hash, una por cada g_k.

Se observa que dos conjuntos A_1 y A_2 son almacenados en la misma celda de la k-ésima tabla si $g_k(A_1)=g_k(A_2)$ para alguna $k=1,2\ldots,l$. Enfatizando que la probabilidad de que dos conjuntos A_1, A_2 coincidan en los r valores Min-hash de una tupla g_k es:

$$P[g_k(A_1)=g_k(A_2)]=simi(A_1,A_2)^r. \tag{15}$$

Entonces, la probabilidad de que dos conjuntos A_1, A_2 tengan por lo menos una tupla idéntica es.

$$P_{colision}[A_1,A_2]=1-(1-simi(A_1,A_2)^r)^l. \tag{16}$$

Recapitulando, lo que hace Min-hashing es generar particiones aleatorias del espacio de tal forma que los conjuntos con similaridad de Jaccard alta son más probables que queden dentro de la misma celda de la partición; lo que significa que este es un método probabilístico que identifica pares de conjuntos semejantes almacenándolos ambos en la misma celda.

3.3. *Sampled Min-hashing* para minado de tópicos

En el año 2015 Fuentes y Meza[5] presentaron una forma de aplicar minado de tópicos con un enfoque distinto a lo que se ha presentado en la sección 2.4 de este trabajo. Este proceso lo hacen utilizando las propiedades de las funciones hash para mapear los "términos semejantes" de una colección de textos y encontrar los conjuntos "parecidos entre ellos". A esta técnica la llamaron *Sampled Min-Hashing* para minado de textos.

Este método no se ha popularizado pero presenta ventajas importantes sobre otros en concepto y en tiempo de cómputo.

Se puede decir que usando la base Min-Hashing se deriva una familia de métodos -*Sampled Min-Hashing* (SMH) y *Sampled Weighted Min-Hashing* (SWMH)-. Estos funcionan bajo los conceptos de co-ocurrencia de términos y agrupamiento de aquéllos que tienen un valor elevado de sobreposicionamiento; estos grupos producen los tópicos de la colección.

La primera hipótesis que maneja este método es que es muy factible que los términos que consistentemente co-ocurren en los mismos documentos pertenezcan al mismo tópico.

Con SMH y SWMH, los tópicos son representados como un subconjunto del vocabulario total, a diferencia de LDA que están representados como una distribución de probabilidad sobre dicho vocabulario.

En 2011 Fuentes Pineda et. al. [20] plantean el algoritmo para el descubrimiento de objetos en grandes colecciones de imágenes. Lo que resalta en su trabajo es que SMH funciona bien para minado de grupos con un valor elevado de su coeficiente de similitud de Jacard. La adaptación de las mismas ideas al entorno de texto se da en dos etapas y es como sigue.

3.4. Etapa I: Partición

Lo primero es definir el modelo de un documento. Al igual que LDA y NMF, SMH y SWMH modelan los documentos como bolsas de palabras, esto significa que no importa en que orden aparezcan, solo importa si están presentes o en su caso, la frecuencia con la cual se encuentran. Lo siguiente por definir son los conjuntos que en secciones anteriores fueron identificados como A_i. En este caso, cada A_i es el conjunto de documentos que contienen un término (palabra) en particular del vocabulario, por ejemplo el termino t_1, y llamaremos T_1 a ese conjunto. De igual forma T_2 será el conjunto de documentos que contienen el termino t_2 y así hasta T_v. A esta estrategia de intercambiar el punto de vista de "conjuntos (documentos) cuyos elementos son los términos que se usan en él" a "conjuntos de documentos que usan un termino en común" se le conoce como lista invertida de documentos y se refiere a cada T_i, como su nombre lo indica, como una lista de documentos.

Ahora se puede generalizar la ecuación (11) como una múltiple similitud de Jaccard nombrada Coeficiente de co-ocurrencia de Jaccard y definida como:

$$JCC(T_1, T_2, ..., T_k) = \frac{|T_1 \cap T_2 \cap ... \cap T_k|}{|T_1 \cup T_2 \cup ... \cup T_k|} \tag{17}$$

Donde el numerador representa la cantidad de documentos en los que los términos t_1, t_2,...t_k co-ocurren y el denominador es la cantidad de documentos con al menos uno de los k-términos. Nótese que k va de 1 *a* v, donde v es la cantidad de palabras en el vocabulario; con eso resulta evidente que el cálculo de todas las intersecciones posibles representa un problema para el poder de cómputo.

De igual forma podemos reescribir la ecuación (13) como:

$$P[h(T_1)=h(T_2)=...=h(T_k)]=\frac{|T_1\cap T_2\cap...\cap T_k|}{|T_1\cup T_2\cup...\cup T_k|} \quad (18)$$

De la ecuación (18) se entiende que la probabilidad de que los *k*-términos tengan el mismo valor Min-hash depende de que tan corelacionada esté su co-ocurrencia; es decir, mientras más correlacionados estén, más alta es su probabilidad de tener el mismo valor Min-Hash. Esto implica que los términos que consistentemente co-ocurren en múltiples documentos tienen una alta probabilidad de tener el mismo valor Min-Hash.

De (17) y (18) se obtiene:

$$JCC(T_1,T_2,...,T_k)=P[h(T_1)=h(T_2)=...=h(T_k)] \quad (19)$$

La gran relevancia de (19) es que gracias a las funciones hash se obtiene una forma probabilística de obtener *JCC* de un grupo de conjuntos sin tener que calcular las intersecciones con un método costeable.

Siguiendo la idea de la sección 3.2.3, se calculan *l* tuplas con *r* valores de Min-hash para encontrar conjuntos de términos con tuplas idénticas, los cuales se convertirán en términos co-ocurrentes. Análogamente a las ecuaciones (15) y (16) pero usando *JCC* se obtiene:

$$P[g_j(T_1)=g_j(T_2)=...=g_j(T_k)]=JCC(T_1,T_2,...,T_k)^r \quad (20)$$

$$P[h(T_1)=h(T_2)=...=h(T_k)]=1-(1-JCC(T_1,T_2,...,T_k)^r)^l. \quad (21)$$

Es interesante analizar que la ecuación en (21) es una función con comportamiento estrictamente creciente para cualesquiera valores de $r,l\in\mathbb{N}$. Los valores de *r* y *l* modifican el perfil haciendo más o menos abrupto el cambio por secciones del dominio como se ve en la Figura 3.2.

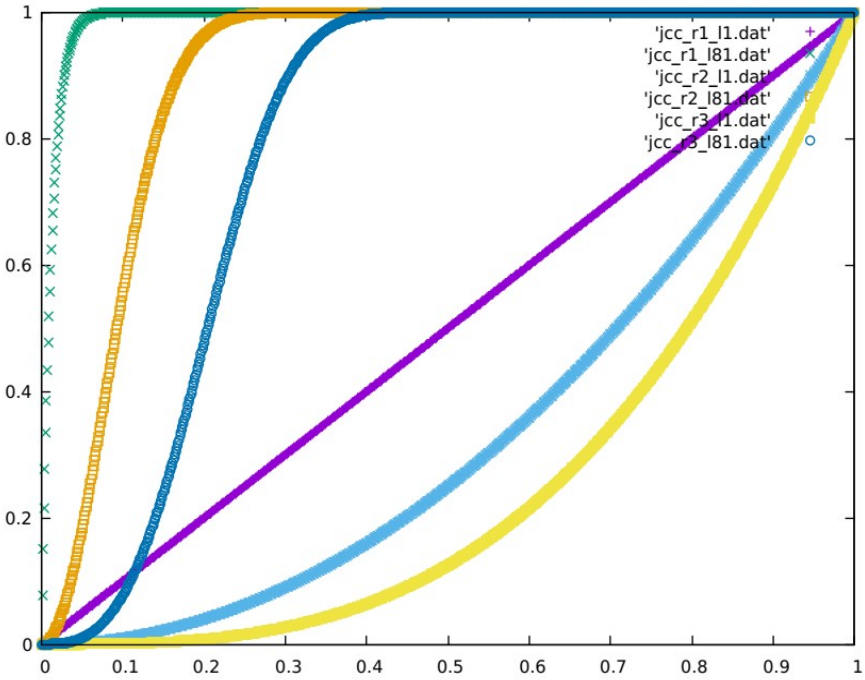


Figura 3.2: Gráfica que muestra el perfil del JCC para distintas combinaciones de valores en r y l.

Ahora, de forma similar a lo expuesto en la sección 3.1 se busca encontrar aquellos conjuntos de *k* elementos que tengan un *JCC* alto. Para lograr ello se tienen que encontrar valores plausibles de *r* y *l* tales que "la probabilidad de que un grupo de *k* términos tenga una tupla idéntica" se pueda aproximar como una función escalón unitaria en cierto valor η que dependerá de "que tan estricto" se quiere el parecido, esto es:

$$P_{colisión}[T_1, T_2, \ldots, T_k] \approx \begin{cases} 1 \, si \, JCC(T_1, T_2, \ldots, T_k) \geq \eta \\ 0 \, si \, JCC(T_1, T_2, \ldots, T_k) < \eta \end{cases} \tag{22}$$

Donde a η se le llama parámetro de similitud y los valores que se pueden pensar para éste son entre 0.2 y 0.8 dependiendo de la aplicación. Un valor típico es 0.5 pero es un parámetro que se tiene que entonar para cada caso. En la siguiente sección se

3.4.1. Selección de parámetros.

El primer valor que se tiene que dimensionar es *r*. Recordando que *r* es el tamaño de las tuplas $g(T_i)$. Es decir, la cantidad de valores Min-Hash que se utilizan en cada $g(T_i)$. Los valores que se han utilizado con buenos resultados son entre 2 y 6. El siguiente parámetro que se dimensiona es η, como éste es el umbral de *JCC* a partir del cual se quiere que los conjuntos con ese valor o mayor tengan una probabilidad cerca a uno de almacenarse en la misma celda se tiene que considerar en relación a los documentos, si son extensos se puede considerar un valor entre 0.2 y 0.4, si son cortos el valor puede aumentar hasta 0.7. A continuación se presentan algunas gráficas del comportamiento de (21) para algunas *r* y *l.*

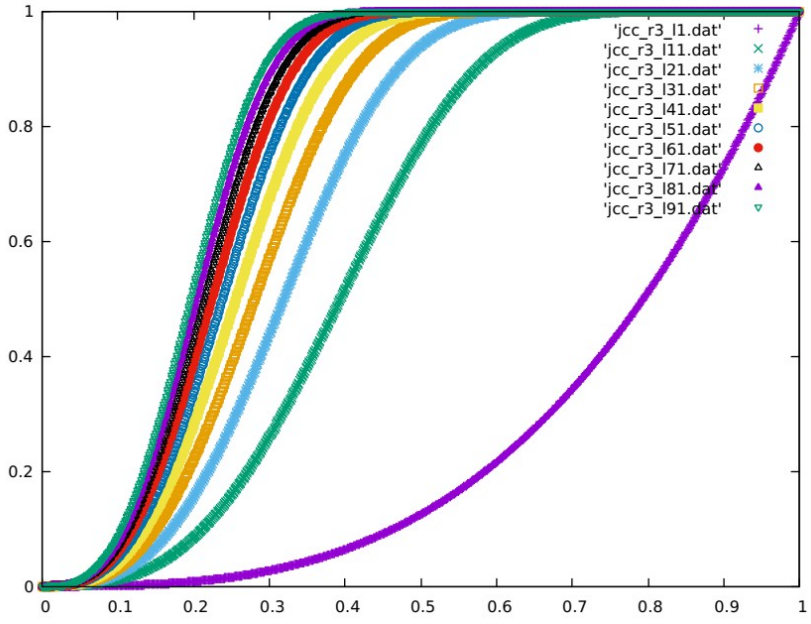


Figura 3.3: Gráfica que ilustra el cambio del perfil de JCC con r constante y distintos valores de l.

Dados η y r se puede determinar l si se centra la gráfica asignando $P_{colisión}[T_1, T_2, \ldots, T_k] = 0.5$ para el valor seleccionado de η; Lo cual resulta en:

$$l = \frac{\log(0.5)}{\log(1-\eta^r)} \tag{23}$$

Una vez establecidas estas bases, lo que se sigue para realizar el minado de tópicos es calcular de forma análoga las *l* tuplas descritas en la ecuación (14) para todos los conjuntos T_j con *j=1,2,..,v* y usando *JCC* en vez de simi. Después se construyen las *l* tablas hash, y se guarda cada lista T_j en la celda correspondiente a $g_i(T_j)$. Si hay colisión en alguna celda de tres o más T_j, entonces las listas que la ocasionaron se almacenan juntas y se extraen para identificarlas de manera especial, a estos se les llama conjuntos de términos co-ocurrentes, en particular, "conjuntos de palabras co-ocurrentes" (CPC). Hasta aquí se puede decir que termina la primera etapa del método llamada etapa de partición.

3.5. Etapa II: Agrupamiento.

Lo siguiente se considera una segunda etapa, la etapa de agrupamiento. En esta se ensamblan los tópicos finales. Como resultado de la etapa anterior se espera que los términos discriminatorios y los términos estables pertenecientes a un tópico queden superpuestos dentro de las celdas de las particiones. Entonces, lo que se hace es unir los *CPC* que comparten "suficientes" palabras para formar los tópicos finales.

Para decidir como hacer estas uniones, se puede medir la proporción de términos compartidos entre dos conjuntos de términos co-ocurrentes C_1 y C_2 a través de su coeficiente de superposición definido como:

$$ovr(C_1, C_2) = \frac{|C_1 \cap C_2|}{min(|C_1|, |C_2|)} \in [0,1]. \quad (24)$$

El siguiente paso, la etapa de aglomeración[5], es calcular el coeficiente de superposisión de cada par de términos, pero como un par de conjuntos de términos co-ocurrentes con similitud de Jacard elevada también tiene un coeficiente de superposisionamiento elevado, entonces, al haber usado min-hashing se evita este calculo para todas las parejas de conjuntos con términos co-ocurrentes.

Se puede decir que en su fase de agrupamiento, el algoritmo combina cadenas de conjuntos de términos co-ocurrentes con un coeficiente de sobreposicionamiento alto para interpretarse como un mismo tópico.

Esto hace que los conjuntos de términos co-ocurrentes asociados al mismo tópico puedan pertenecer al mismo grupo, mientras sean de la misma cadena, aunque no compartan términos con otro. Finalmente los grupos formados tienen la propiedad de que para cualquier conjunto de términos co-ocurrentes existe al menos otro conjunto de términos co-ocurrentes en el mismo grupo que tiene un termino de superposiciónamiento más alto que un cierto umbral E.

3.6. Samples Weighed minhashing

Sample weighed minhashing es una mejora para *SMH* en la que se considera la extensión del texto en donde aparecen los términos dándole más peso a los términos que aparecen en documentos cortos. La forma que se hace esto es usando la ecuación (12) en vez de (11) para obtener una generalización análoga a la obtenida para *JCC* denominada ahora coeficiente de coocurrencia ponderada (*WCC*) como sigue:

$$WCC(T_{1,..}, T_k) = \frac{\sum_i w_i min(T^i_1, ..., T^i_k)}{\sum_i w_i max(T^i_1, ..., T^i_k)} \in [0,1] \quad (25)$$

Donde $T^i_1, ..., T^i_k$ son las frecuencias en las que los términos $T_1,..,T_k$ ocurren en el i-ésimo documento y w_i está dado por el inverso del tamaño del i-ésimo documento. A continuación se resume el algoritmo:

Algoritmo: Minado de tópicos usando SWMH
Datos: Lista invertida de documentos $T = T_1,\ldots,T_N$. **Resultado:** Tópicos minados $O_1,...O_M$. **Etapa I: Particionado.** 1. Calcular l túplas MinHash $g_i(T_j), i=1,..,l$ para cada lista $T_j, j=1,\ldots,N$ *en T.* 2. Construir l tablas hash y almacenar cada lista $T_j, j=1,\ldots,N$ en la celda correspondiente $g_i(T_j), i=1,..,l$. 3. Marcar cada grupo de listas almacenados en la misma celda como como un conjunto de termino co-ocurrente **Etapa II: Agrupamiento.** 1. Encontrar parejas de conjuntos de términos co-ocurrentes con un coeficiente de superposisionamento mayor a un umbral elegido ϵ. 2. Elaborar una grafica G con los conjuntos de términos co-ocurrentes como vértices y las aristas definidas entre los pares con coeficientes de superposisionamiento mayor a ϵ. 3. Marcar cada elemento conectado de G como un tópico.

Tabla 3.1: Algoritmo para minado de tópicos usando SWMH.[11]

11 Traducido de [5]

3.7. Visualización minHash

El objetivo de este trabajo es visualizar los tópicos que minHashing puede encontrar en las tesis UNAM de las carreras de computación y de ésta forma observar como se van desarrollando y entrelazando los mismos a lo largo del tiempo.

Para realizar la visualización se sigue una estrategia similar a la empleada en la etapa II descrita en la sección 3.5. El corpus se divide en intervalos ordenados de tiempo y se efectúa un minado de tópicos para cada intervalo teniendo así tantos modelos de tópicos como intervalos temporales se eligieron. Posteriormente se considera cada tópico como un nodo de un grafo *G*. Este grafo solo tiene aristas salientes de un nodo ubicado en un intervalo de tiempo y entrantes en un nodo del intervalo subsecuente. Para obtener las aristas se evalúa el coeficiente *overT* para todas las combinaciones de tópicos de cada año con todos los del año subsecuente descrito en la ecuación (26) y se marcan como conectados aquellos pares que rebasan un umbral β elegido.

$$overT(A,B)=\frac{(A\cap B)}{max(|A|,|B|)} \tag{26}$$

Se tiene que remarcar que en la ecuación (26) *A* y *B* representan dos conjuntos de términos a los que llamamos tópicos. Un hecho importante en el cálculo de *overT* es que durante el minado de tópicos, el algoritmo guarda la cantidad de veces que un termino coocurre entregando los resultados a manera de lista con el termino y las repeticiones asociadas a este. Para *overT* cada repetición de un mismo termino se considera un elemento de el conjunto por lo cual los entes con los que se trabaja son conjuntos **con** repetición de elementos. De esta manera si dos conjuntos coinciden en pocos términos pero estos tienen

un alto numero de repeticiones en cada uno, entonces es posible que se rebase el umbral β designado y entonces se podrá marcarlos como conectados. Cada grupo conectado en este grafo se considera el mismo tópico a través de los años.

Capítulo 4: Desarrollo y Experimentación.

En esta sección se describe el desarrollo que se siguió para cumplir las metas plateadas en la sección 1.2 y se plasma en la siguiente linea de trabajo:

1.1 Levantar el *Docker* específico para min-hashing en el servidor Markov de IIMAS y establecer una conexión *ssh* para poder trabajar en él.

1.2 Descargar las tesis de la página Tesis UNAM en Markov.

1.3 Separar y almacenar los documentos por años.

1.4 Hacer el pre-procesamiento de datos para pasar de formato PDF a un TXT adecuado.

2.1 Implementar rutinas de programación que permitan realizar el minado iterativamente para los distintos años.

2.2 Implementar rutinas de programación que permitan comunicarse con el sistema ya implementado para controlar los parámetros del minado.

3.1 Realizar el minado de tópicos para distintos modelos (todos los años y con distintos parámetros).

3.2 Verificar composición de tópicos.

4.1 Implementar una rutina que calcule la relación entre los tópicos de un año y el siguiente.

4.2 Implementar una visualización en diagramas *sankey.*

4.3 Implementar una visualización para las relaciones en forma de grafo.

4.4 Analizar modelos.

4.1. Adquisición y preprocesamiento de datos

Para este trabajo se seleccionó el algoritmo *SWMH* sobre *SMH* para el análisis debido a que los documentos de tesis son muy extensos y resulta relevante el pesado de los términos. Se estudiaron los resultados que se encuentran al ejecutar *SWMH* sobre el subconjunto de tesis UNAM comprendido como las tesis relacionadas con las carreras de cómputo.

Lo primero que se hizo fue navegar el sitio de consulta tesis.unam.mx, ahí se observó que el sistema Oreon permite varias opciones de búsqueda, esto fue el primer factor que determinó el subconjunto de trabajo. Se limitó la búsqueda “por carrera” y con la “palabra” computación. Esto con el entendido que entrega todas las tesis de alguna carrera con la palabra computación en el nombre de ella, por ejemplo “ingeniería en computación”, “ciencias de la computación”, etc...

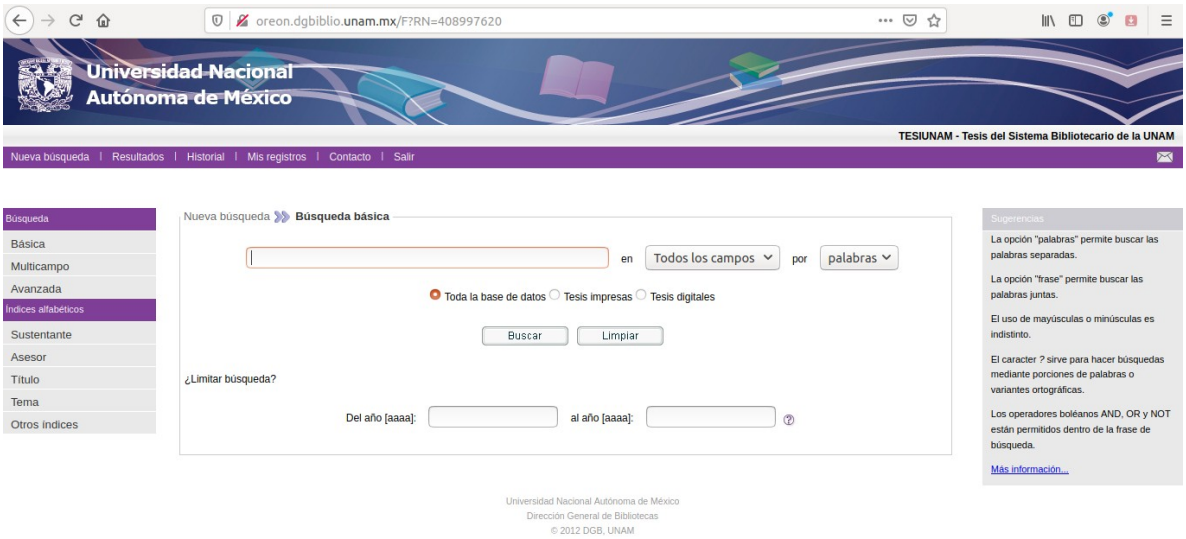


Figura 4.1: Toma de pantalla de la apariencia del sistema en linea Oreon.

Después se observó la calidad de los archivos y se evaluo cuales funcionan. Lo que se encontró es que a partir del año 2004 las tesis están digitalizadas y pasadas por un *OCR* suficientemente bueno tal que el texto digital se ve coherente y legible. A partir del 2009 la calidad de las tesis mejora mucho debido a que ya son digitales desde el origen. El anterior hecho acotó el rango de años para trabajar a 14, de 2004 a 2018. La búsqueda entregó un total de 2,855 registros.

El siguiente gran paso fue obtener los documentos, para ello se utilizó un "*crawler*" previamente programado en javascript por el estudiante Marco Godinez Bustos con otros fines (https://github.com/markotom/crawler-theses), el cual es capaz de obtener las *URL* de cada PDF que entrega la búsqueda y guardarlo en una base de datos MongoDB. El primer problema que se encontró es que las *URL* no son absolutas sino que entrega un *token* que caduca. Después se programó en python un script para leer las *URL* de esa MongoDB, hacer un ajuste a las *URL* para superar el problema del *token*, con eso llegar al lugar real donde están guardados los PDFs y poder descargar las tesis. El script se tiene que correr con acceso a la MongoDB y se llama -descargar.py- (fuera del docker).

Cada tesis mide 10Mb aproximadamente y son 2,855 registros entonces se está hablando de más de 28Gb a descargar. Para poder hacerlo y manejar estos datos se montó un docker en el servidor Markov del IIMAS desde donde posteriormente se realizó todo el preprocesamiento de datos y los cálculos de SWMH.

Una vez obtenidos los PDF se presentó un imprevisto adicional, los archivos tal cual como se descargan están protegidos. Esta caracteristica impide que se puedan transformar a texto plano lo cual es necesario para el procesamiento de los textos. Para sortear el inconveniente se realizó otro script llamado -unlock.py- en python que utiliza el programa de línea de comando qpdf el cual puede hacer la tarea requerida y antes mencionada en todos los documentos.

Con los documentos PDF desbloqueados se pudo seguir a transformarlos a texto plano, aquí se utilizaron dos métodos para evaluar si alguno de ellos entregaba mejores resultados. El

primero fue el pdftotxt y el segundo fue un script en python con la librería pdfminer llamado -pdf2txt.py-. Con una inspección manual no se pudo encontrar diferencia en los resultados, ambos presentan un problema con los acentos de los documentos pasados por el OCR, la mayoría de las veces los identifica bien pero un pequeño porcentaje corta la palabra en dos, el lugar del caracter con acento lo interpreta como un espacio. Al ser los resultados de ambos métodos casi iguales, se decidió quedar con los obtenidos por el script con pdfminer.

Todos los documentos se organizaron en carpetas por año. Posteriormente, siendo que cada documento es extenso, una estrategia que se decidió tomar fue dividir cada tesis en 6 partes, así tendremos más documentos para analizar y también se da la oportunidad de encontrar un tópico dentro de una sola tesis. Esto se hizo con un script sencillo en python. Finalmente con esto se tiene un corpus de poco más de 17,000 documentos.

4.2. Procesamiento de datos con *SWMH*.

El procesamiento de datos se hizo utilizando el algoritmo programado por el Dr. Gibran Fuentes y el Dr. Ivan Meza ubicado en el repositorio https://gitlab.com/ivanvladimir/smh_docker. Dicho algoritmo se corre con 3 procesos:

1. dir2corpus.py transform
2. discover.py process
3. discover.py explore

Para lograr el objetivo se hicieron 2 nuevos scripts. El primero - complete_model.py - es para integrar en un solo script los pasos ya programados en el repositorio antes descritos y aumentar la importe funcionalidad de realizar el cálculo año por año y para distintos valores de r y l.

El segundo script sirve para obtener las relaciones entre los tópicos de años distintos y graficarlas. Es en este es donde se centraron la mayor parte de las decisiones importantes.

4.2.1. Tópicos como cuentas

La primera decisión fue: usar los tópicos encontrados como conjuntos sin repetición o como conjuntos con repeticiones. El algoritmo de *SWMH* entrega tópicos con la información de cada palabra que lo compone y la cantidad de veces que esta coocurrió como se ve en el ejemplo:

wigner:144 doppler:135 ville:134 michoacán:131 jalisco:126 tamaulipas:125 mutación:121 nox:118 arteria:115 pimf:111 ultraónica:111 websphere:108 fft:107 weblogic:104 genético:101 chihuahua:99 tabasco:99 yucatán:97 nucleótidos:94 sanguíneo:94 línico:93 retropropagación:93 votar:92 genéticos:91 ultrasónicas:88 ultrasónico:88 cromosoma:87 neuronal:87 verlag:87 neuona:85 disca:83 neuronas:83 born:81 oaxaca:80 perceptrones:80 electorales:78 springer:78 genómica:77 electores:76 meteoroógicas:76 medica:75 molécula:74 colima:73 votaciones:73 padecimientos:72 querétaro:72 siirfe:72 genética:71 clínicos:70 drango:70 tlaxcala:70 carbono:68 padecimiento:67 simulada:67 cruzamiento:66 ecología:66 cromosomas:65 morelos:65 ozono:65 conv:64 genoma:64 espectral:63 electoral:62 atmosféricas:61 elector:60 nitrógeno:60 pacientes:60 workshop:60 patológicos:59...

Tabla 4.1: Ejemplo de un tópico truncado

Las primeras palabras coocurren más veces y va decreciendo hasta llegar a solo una coocurrencia (no se ve en el ejemplo porque está truncado el tópico), una forma de interpretarlo es “que tan importante es cada palabra en el tópico”. Si consideramos el tópico sin repetición entonces esta información se pierde y todas las palabras tienen el mismo peso, es por ello que se tomó la decisión de considerar estos conjuntos **con** repetición. En el script -FINAL_graficar_contador.py- esto se hace con un “counter” de la librería “collections”.

4.2.2. Uso de *stopwords*

El código de *SWMH* tiene incluido un parámetro importante que se le llama "*stopwords*", este parámetro sirve para que el algoritmo que genera el vocabulario no considere un listado de palabras. En este caso se incluyeron palabras de uso muy común y repetitivo del idioma español como pueden ser: hizo, muy, podrían, el, la, los, etc. Para esto se usó como base el documento propuesto en https://github.com/stopwords-iso/stopwords-es/blob/master/stopwords-es.txt.

Después de los primeros resultados se observó que el problema de los acentos originado por el OCR introduce muchas palabras de las anteriormente mencionadas como *stopwords* pero divididas, el caractér con acento algunas veces no se reconoce adecuadamente y se interpreta como un espacio. Lo que se hizo fue tomar un tópico que incluía varias de éstas palabras divididas y agregarlas una por una al listado de *stopwords.*

4.2.3. Umbral de correlación

Una parte importante del proyecto es encontrar cuando se consideran vinculados dos tópicos de años subsecuentes. El acercamiento que se eligió para ver la evolución de los tópicos en el tiempo fue un modelo directo el cual está descrito en la Sección 3.7, este se hizo con la misma idea del *Overlaping* desarrollada en la sección 3.5 solo que adaptado al formato de cuentas en vez de conjuntos sin repetición.

La idea es encontrar la intersección de cuentas entre cada uno de los tópicos de un año y cada uno de los tópicos del siguiente año, y después, dividir este resultado entre el tamaño de el conjunto con mayor cantidad de cuentas, con esto se consideró que dos tópicos están relacionados cuando el coeficiente *overT* antes mencionado es mayor a 0.1. La forma de asignar este valor fue empírica. Se observaron tres parejas de tópicos formadas por un

tópico de un año y otro del año subsecuente que para mi juicio como humano parecen estar relacionados y se calculó el *overT de cada par relacionado*, de esta forma me di una idea de los valores que tienen que estar incluidos para considerar que dos tópicos están relacionados y utilicé el valor mínimo de los 3 para tomarlo como umbral de relación. Sin duda tiene que existir algún mejor criterio y esto se propone como trabajo a futuro. Me parece importante resaltar que se propuso un procedimiento sencillo el cual entrego resultados plausibles, es por ello que no hubo necesidad de regresar y replantear el método.

4.3. Métricas.

Para evaluar los resultados se plantearon los siguientes parámetros como métricas de los modelos.

1. El tamaño total del tópico más chico en el modelo (contando las repeticiones de todos los elementos).
2. El tamaño total del tópico más grande en el modelo (contando las repeticiones de todos los elementos)
3. El tamaño promedio de los tópicos en el modelo (contando las repeticiones de todos los elementos)
4. Cantidad máxima de palabras distintas en un tópico.
5. Cantidad mínima de palabras distintas en un tópico.
6. Cantidad promedio de palabras distintas en un tópico.
7. Número de tópicos por modelo.
8. Media

También se agregaron métricas basadas en las propiedades del grafo tales como:

1. **Grado Promedio**. - Cantidad promedio de aristas por nodo.

   Numero de aristas/numero de nodos

2. **Diámetro de la red. -** La distancia máxima entre cualquier par de nodos en un grafo.

3. **Densidad de la gráfica**- Un gráfico denso es un gráfico en el que el número de aristas está cerca del número máximo de aristas. Lo contrario, un gráfico con solo unas pocas aristas, es un gráfico disperso. La distinción entre gráficos dispersos y densos es bastante vaga y depende del contexto.

   La densidad de gráficos simples se define como la relación entre el número de aristas | E | con respecto a las máximas aristas posibles. [12]

4. **Modularidad** – Es una forma de medir la fuerza de la división de una red en módulos (también llamados grupos, clústeres o comunidades). Las redes con alta modularidad tienen conexiones densas entre los nodos dentro de los módulos, pero conexiones escasas entre nodos en diferentes módulos.[13]

5. **Elementos conectados fuerte y débil**. Un gráfico dirigido se denomina débilmente conectado si al reemplazar todos sus aristas dirigidas con aristas no dirigidas se produce un gráfico conectado (no dirigido). Está conectado unilateralmente o unilateral (también llamado semiconectado) si contiene una ruta dirigida de *u* a *v* o una ruta dirigida de *v* a *u* para cada par de vértices *u, v*. Está fuertemente conectado, o simplemente fuerte, si contiene un camino dirigido de *u* a *v* y un camino dirigido de *v* a *u* para cada par de vértices *u, v;*[14] es decir, una gráfica está fuertemente conectada si cada vértice es accesible desde cualquier otro vértice. Los componentes fuertemente conectados de un grafo dirigido arbitrario forman una partición en subgrafos que están ellos mismos fuertemente conectados.[15]

12 Definición obtenida de https://en.wikipedia.org/wiki/Dense_graph
13 Definición obtenida de https://en.wikipedia.org/wiki/Modularity_(networks)
14 Obtenido de https://en.wikipedia.org/wiki/Connectivity_(graph_theory)
15 Obtenido de https://en.wikipedia.org/wiki/Strongly_connected_component

4.4. Experimentos.

En esta sección se detalla como se escogió el modelo más adecuado de entre los nueve.

4.4.1. Selección de modelo

Se implementaron 9 modelos con las combinaciones de parámetros *r=1, 2, 5* y *l=200, 300, 400* y se estudiaron para encontrar cual de ellos entrega los resultados más plausibles con base en las métricas antes mencionadas.

Los valores obtenidos para estas métricas se condensan en las siguientes tablas:

		Promedio sobre todos los años								
r	*l*	Tópicos encontra-dos	Máximo de cuentas	Mínimo de cuentas	Máximo de palabras	Mínimo de palabras	Promedio de cuentas	Promedio de palabras	Media de cuentas	Media de palabras
2	200	635	68887	14	7620	5	236	75	51	32
2	300	1399	212668	12	11762	4	234	55	46	29
2	400	2477	382243	11	12256	3	227	47	46	28
3	200	119	12416	13	900	5	276	63	46	25
3	300	166	25321	12	1753	4	359	71	43	25
3	400	223	45036	12	3110	4	412	74	42	25
5	200	22	4698	13	377	5	496	52	49	15
5	300	28	9843	11	513	3	664	51	42	14
5	400	37	15436	11	622	3	755	49	41	14

Tabla 4.2: Métricas estadísticas para los modelos con r=2,3,5 y l=200,300,400

r	*l*	Grado promedio	Diámetro de la red	Densidad de la gráfica	Modularidad y comunidades encontradas resolución 8		Strong connected	Weakly Connected Components	Av path leng
2	200	0.782	12	0.000	0.94	555	1823	553	4.05
2	300	0.800	14	0.000	0.98	1329	5157	1327	2.52
2	400	0.932	14	0.000	0.93	1987	12100	1981	3.07
3	200	0.935	14	0.002	0.89	162	583	161	4.69
3	300	1.167	14	0.002	0.8	216	778	213	4.69
3	400	1.736	14	0.002	0.64	256	1048	255	5.99
5	200	1.087	14	0.007	0.77	23	149	22	4.71
5	300	1.143	14	0.006	0.69	36	203	36	4.56
5	400	1.412	14	0.006	0.74	34	250	33	5.20

Tabla 4.3: Métricas de Grafo para los modelos con r=2,3,5 y l=200,300,400

Año	Tamaño del vocabulario
2004	21309
2005	16569
2006	9670
2007	9294
2008	15099
2009	14830
2010	11778
2011	13219
2012	11066
2013	11086
2014	10670
2015	11060
2016	8999
2017	8446
2018	6781
Promedio	**11992**

Tabla 4.4: Tamaño de los vocabularios de cada año.

Un valor que salta a la vista en la Tabla 4.3: métricas de grafo, es el diámetro de la red en la que para todos los casos, excepto *2r-l200* es 14; lo cual coincide con la cantidad de años estudiados (2004-2018), eso quiere decir que, como los tópicos se conectan en años sucesivos y el diámetro de la red es la distancia mas corta entre los dos nodos mas distantes y el grafo es dirigido, entonces por lo menos un tópico se mantiene conectado todo el tiempo.

Una métrica que llama la atención es la cantidad promedio de tópicos por año que se encuentra en los modelos *r2-l300* y *r2-l400* siendo arriba de mil y dos mil respectivamente. Si se considera que son menos de 3,000 tesis las que se analizan en todo el corpus promediando alrededor de 200 tesis por año entonces estaríamos hablando que en promedio encuentra entre 7 y 10 tópicos por tesis. Esas cantidades de tópicos pueden ser plausibles ya que las tesis son documentos extensos y tocan varios temas pero en este análisis me gustaría enfocar en menos tópicos buscando los más relevantes, por ello dejaré de lado los modelos *r2-l300* y *r2-l400*. Con esta misma línea de pensamiento *r2-l200* es un buen candidato ya que entrega en promedio 3 tópicos por tesis. Esta cantidad es comparable con la cantidad de palabras clave que se solicitan en el registro de los artículos científicos y por ello considero que es un buen número.

Los modelos con *r5* parecen ser muy estrictos, de una forma opuesta a lo que pasa con los modelos con *r2* porque encuentran menos de 40 tópicos en promedio por año que representa menos de un tópico por tesis por año. Esto podría ser interesante porque solo ve las coocurrencias más altas (en 5 documentos) esto quiere decir que encuentra tópicos muy “de moda” para ese año, lo cual podría ser información valiosa para otro tipo de análisis pero para un punto de vista global pierde mucha información y no me parecen elecciones adecuadas; además, estos tópicos que se forman pueden estar viciados por palabras de uso común que no se incluyeron en las *stopwords* y que se repiten en muchos documentos haciendo estos tópicos poco atractivos. Una alternativa que se puede intentar en el futuro con estos modelos es elevar el valor de *l* y observar que tipo de resultados entrega.

Analizando las métricas del modelo *r3-l400* noté que muestra 223 tópico en promedio por año, que resulta poco más de un tópico por tesis por año, aunque es un número bajo, me parece bueno ya que se puede pensar que encuentra el tópico más relevante de cada tesis.

En este punto podemos centrar la atención en los modelos *r2-l200* y *r3-l400*. Para ver las diferencias entre estos se puede calcular el cociente de cada una de las métricas respectivas, con ello se puede ver que las métricas son muy parecidas, donde difiere considerablemente es en número de tópicos encontrados (casi el triple uno que el otro) y en el máximo de palabras (aproximadamente 2.5 veces más uno que el otro), para todo lo demás son similares y no podría decantarme por alguno.

Observando la Tabla 4.3 encontramos algunos valores sobresalientes para el modelo *r3-l400*, en él se encuentra el camino promedio más largo en el grafo y tiene el grado promedio más alto entre todos los modelos. Al mismo tiempo encuentra 257 comunidades de tópicos lo cual tiene un impacto importante ya que es el que tiene la modularidad más baja.

Con estos argumentos puedo seleccionar el modelo *r3-l400* como el más adecuado para detallar resultados.

4.4.2. Resultados de *r3-l400*.

Para estudiar los modelos se realizó un script capaz de hacer dos cosas, la primera es acomodar la información para mostrar una gráfica tipo *sankey*[16] como la que se muestra en la Figura 4.2.

16 https://es.wikipedia.org/wiki/Diagrama_de_Sankey

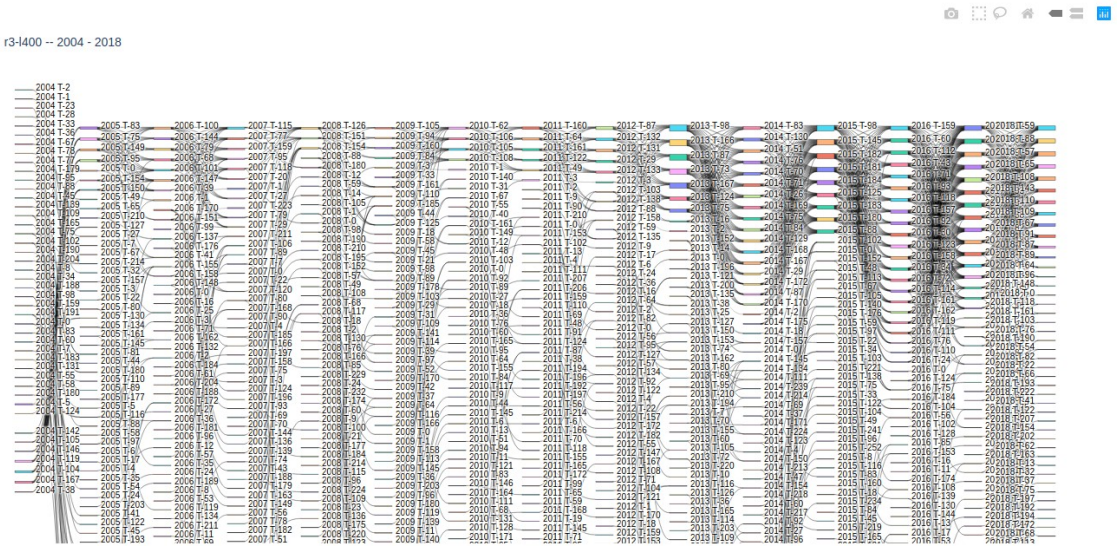


Figura 4.2: Ejemplo de mala visualización de un diagrama sankey para r3-l400 de 2004 a 2018

Por la gran cantidad de nodos y conexiones no es posible visualizar bien el comportamiento, incluso no caben todos los tópicos de cada año en el marco de graficación. Esa es una de la razones por las que la segunda función del script es exportar los datos a formato GEXF usando la librería networkx de python para después visualizarlo en la aplicación Gephi. También, fue en esta aplicación que se calcularon las métricas de grafo para todos los modelos.

A continuación muestro una visualización del modelo *r3-l400* con separación por módulos. Cada nodo representa un tópico de algún año y cada arista representa una relación encontrada entre los dos nodos en cuestión. Dicho coloquialmente, cada color representa un tópico que se encontró en por lo menos dos años consecutivos.

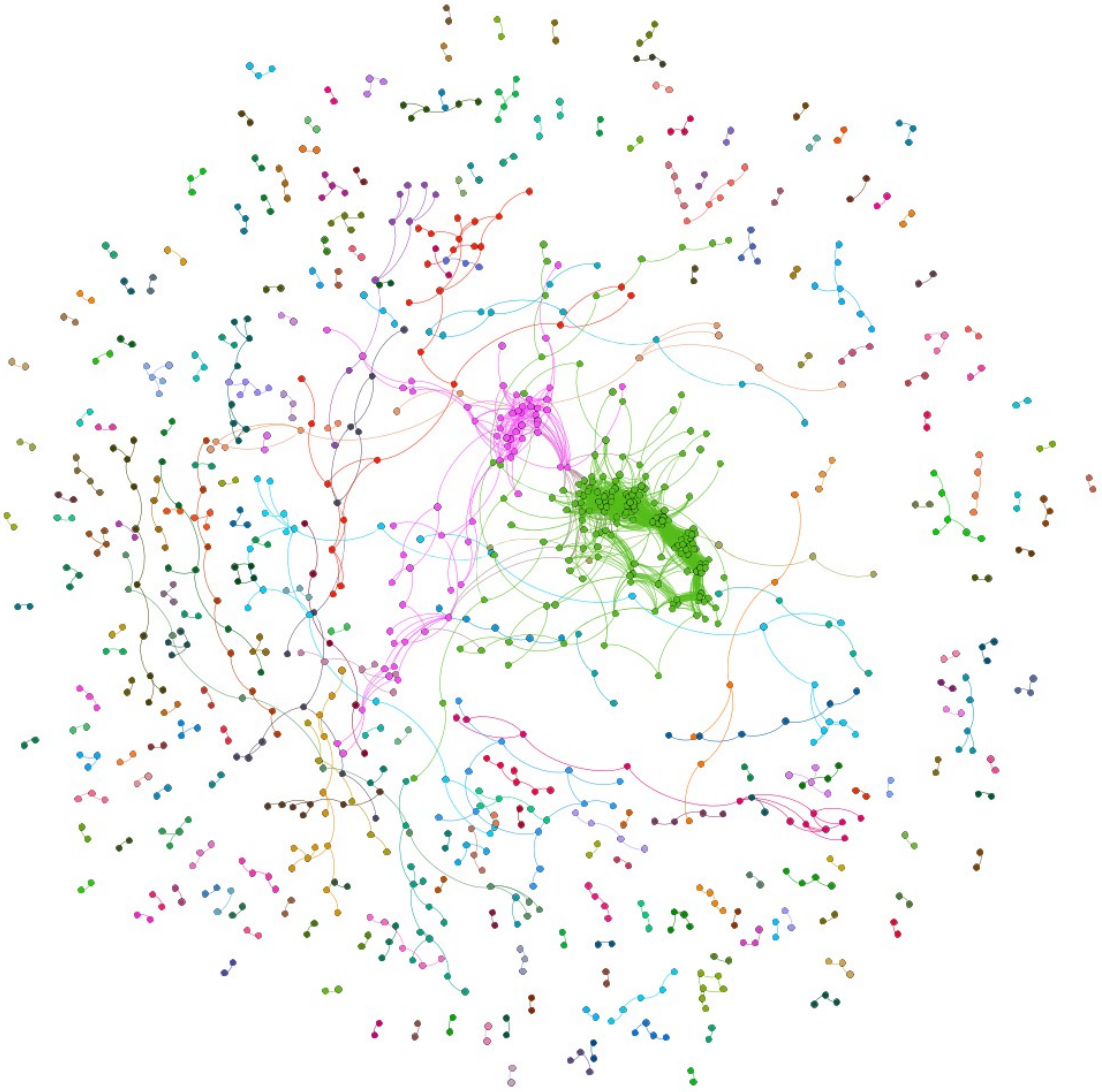


Figura 4.3: Grafo completo de tópicos conectados en el tiempo.

Los dos grupos más grandes (verde y rosa) son los tópicos más conectados, sin embargo al revisar directamente las palabras que los componen parece que no otorgan información interesante porque están tan conectados con todo que es imposible encontrar un patrón. Dejando un de lado estos dos grupos podemos analizar los siguientes grupos más grandes

(naranja y azul). El grupo naranja parece poco interesante porque contiene solo palabras en inglés las cuales no puedo ubicar, por ejemplo:

160 same:6 without:6 only:4 keep:2 rule:2 root:1

Tabla 4.5: Ejemplo de un tópico.

En la Tabla 4.5 se muestra como entrega el método un tópico. El primer numero que aparece (160) es un identificador que se asigna al numero de tópico en cada año. Los números que aparecen después de cada palabra son las veces que esa palabra coocurrio. En este ejemplo se muestra un tópico “pequeño” -solamente 6 palabras distintas- lo cual no es lo más común.

Con esto llegamos al tópico más grande que podemos interpretar, el azul-claro ubicado al norte de la gráfica.

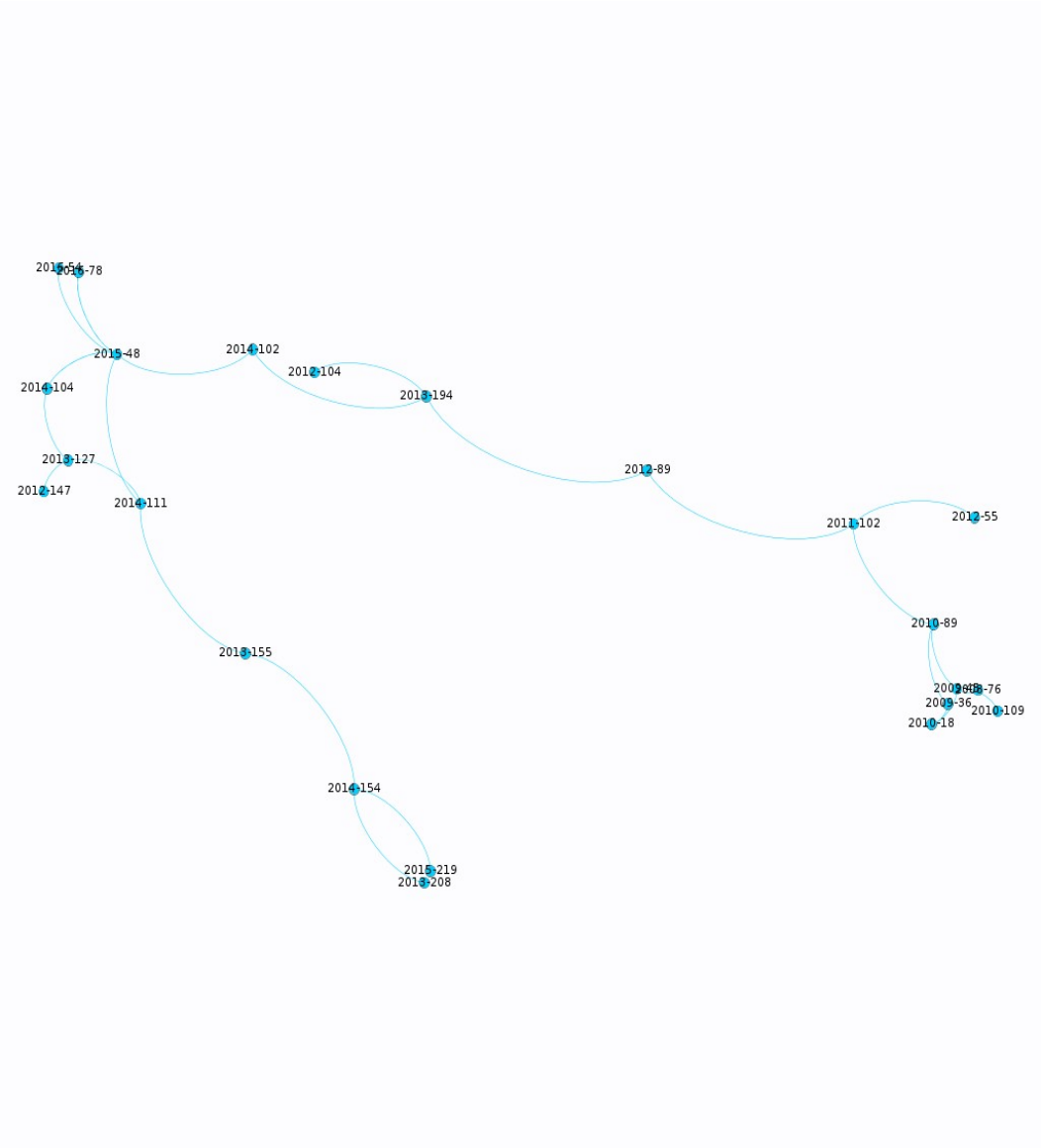


Figura 4.4: Acercamiento al grupo de mayor tamaño con significado (transferencia de datos en red o protocolos de red).

Al revisar los tópicos que lo integran vemos que encuentra un camino desde el 2008 hasta el 2016 y está conformado por el 2.19% del total de nodos en la gráfica. A continuación se presentan los tópicos resumidos, para más información de los mismos se puede revisar el anexo. El identificador del campo semántico se asignó manualmente según mi interpretación. A continuación se presentan algunos de los tópicos que se encontraron:

Tópico azul-claro Figura 4.4: Protocolos de Redes.	
2008	protocolo, protocolos, protocol, paquetes, dispositivos, tcp, access, request, comandos, investigar, ip, recepción, transferencia,...
2009	tcp, ip, protocolos, protocolo, protocol, direcciones, ftp, transferencia, abiertos, access, bloque, comunicaciones, conexiones, enrutamiento, espera, tráfico,...
2010	puertos, tráfico, paquetes, puerto, tcp, banda, bytes, disponibilidad, ethernet, ftp, ip, lan, protocolos, soporta, udp, vía,...
2011	tcp, protocol, tráfico, udp, bytes, network, paquetes, router, address, ataque,..
2012	host, protocol, configuration, ip, tcp, dynamic, hosts, puerto, docs, gigabit, paquete, products, router,...
2013	dns, domain, ipv, cpu, dhcp, host, network, operativos, protocol, remoto, servidores, tcp,..
2014	ancho, banda, medida, conectarse, gateway, mecanismo, números, señales, tcp, telecomunicaciones, transmisión, tráfico, utilización,...
2015	49 tcp:11 ip:10 paquetes:9 protocolo:9 protocolos:8 mensaje:4 protocol:4 capa:3 comparativa:3 networking:3 banda:2 lan:2 local, mac, network, osi, redes, transmite, voz,...
2016	ip, protocolo, protocolos, comunicaciones, adoptar, dispositivo, dispositivos, enlace, evolutivo, falla, fallas, frecuentemente, orientado, programado, protocol,...

Con esto se puede ver que la línea principal mantiene el mismo tono muy enfocado a protocolos de transferencia de datos en red; sin embargo, hay algunos nodos que se van uniendo a esta línea que tienen ligeras diferencias. Por ejemplo:

2012	lan, receptor, router, ceros, emisor, network, banda, fabricantes, gateway, ieee, interconexión, osi,...
2013	dns, domain, ipv, cpu, dhcp, host, network, operativos, protocol, remoto, servidores, tcp, comando, computadora, conectarse, configurar, gateway, hora, instalado, ip, lan, llegue, máscara, nodos, pc, recibe, remota, sub, virtuales.

Los anteriores dos tópicos están relacionados con redes y protocolos de transferencia de datos pero no se ven tan sesgados a esos temas, sin embargo, el algoritmo puede encontrar esta relación y los identifica en el mismo grupo.

Los siguientes módulos en tamaño, se pudieron aislar filtrando en Gephi aquellos que tuvieran entre el 0.9% y 1.5% de los nodos totales. El resultado del filtrado está en la siguiente figura.

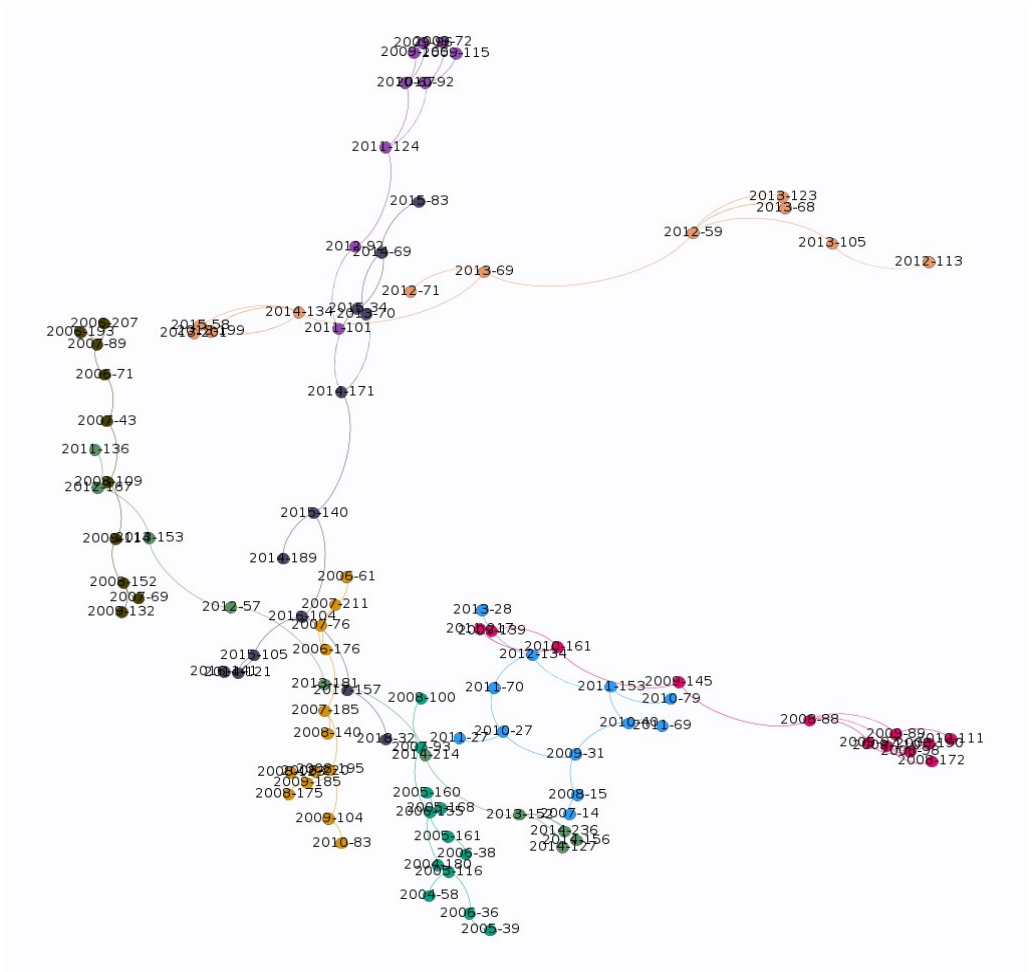


Figura 4.5: Filtrado de módulos entre 0.9% y 1.5% de los nodos totales.

Continuaré esta discusión analizando algunos de los módulos a detalle.

El grupo azul claro de la Figura 4.5 se ve interesante porque está vigente desde 2007 hasta 2013 y se aprecia que en 2009 se separa por dos caminos y más adelante en 2012 se vuelve a unir. Al revisar los tópicos directamente encontramos lo siguiente:

Tópico azul claro Figura 4.5: Criptografía.	
2007	descifrar, cifra, criptográfico, encripción, secreto, criptología, destinatario, secreta, negar, cifrar, criptografía, cifrada, griego, remitente, descifrado, atacante, convirtiendo, criptográficas, vulnerabilidades, asimétrica, encriptado, encriptar, firmas, ocultar, phtml, protegerse,...
2008	cifrada, aes, pretty, ipsec, pgp, descifrar, good, cifrar, descifrado, cipher, cifrados, encryption, privacy, rsa, cifrado, force, secreta, secure, ietf, kerberos, task, túneles, xor, cifradas, túnel, autentica, autenticado, autenticidad, cbc, certificado, cifra, ssl, vpn, atacante, authentication, certificados, des, dsa, intruso, parches, spoofing, tls, criptográficas, criptográficos, garantizando, inseguros, intrusiones, llave, misiones, pda, pki, robusto, sas, sha, smtp, vulnerar,...
2009	criptografía, criptográfico, cifrar, repudio, descifrado, descifrar, concientización, asimétrica, criptográficas, dañar, destruida, cifrado, asimétrico, autorizada, desastres, devolución, ids, posesión, atacantes, ataques, autenticación, autenticidad, cifrados, contingencia, hash, incumplimiento, simétrica, vulnerabilidades,...
2010	criptografía, descifrado, ocultar, cifrar, secreto, criptoanálisis, descifrar, autenticación, protege, repudio, secreta,...
2011	secreto, descifrar, secreta, descifrado, secretos, cifrar, hellman, llaves, romper.
2012	asimétrica, receptor, cifrar, descifrar, emisor, intercepción, simétrica, activos,...
2013	cifrar, cifrado, rsa, secreta, descifrado, descifrar, atacante, autentificación, cifrada, encriptación, cifra, criptografía, criptográfico, privacy, cifradas, descifra, inseguro, públicas, robo, segura, simétrica,...

En este caso parece claro que el tema principal del tópico es **criptografía**.

Otro ejemplo que podemos ver es el grupo morado que empieza desde varios puntos en 2009 y encuentra un camino hasta el 2012. Una muestra de lo que encontramos es lo siguiente:

Tópico morado Figura 4.5: Seguridad Informática.	
2009	ids, nids, intrusos, ips, intrusiones, honeypots, vulnerabilidades, alerta, ataque, firewalls, gusanos, intrusión, snort, trafico,...
2009	ataques, cifrado, ataque, legítimo, protocolo, vulnerabilidades, atacante, auditoría, intercepción, intruso, pasivos,...
2009	nmap, intrusion, nessus, ips, atacante, nids, activamente, alerta, banderas, coincidencia, exploits, msg, subredes, syn, trafico.
2009	ataques, autenticación, autorizado, autorizada, ataque, auditoría, autorizados, denegación, intrusiones, vulnerabilidades,..
2010	repudio, autorizada, autorizados, confidencialidad, secreto, amenazas, auditoría, autorizadas, ataque, autorización, desastres, políticas, proteger, amenaza, débil, hackers, intruso, nsa, política, robar, united, virus, vulnerabilidad.

2010	ids, detection, intrusion, intrusos, malicioso, contraseñas, proteger, alertas, amenazas, capaces, considerarse, dejar, disco, firewall, firmas, identidad, instante, lógica, motivos, packet, seguramente, snort, subir, virus.
2011	antivirus, autorizado, malicioso, vulnerabilidades, autenticación, confidencialidad, filtrado, parches, prevención, security, sesión, virus,...
2012	prevención, ataques, ataque, amenazas, autorizado, confianza, firewall, informáticos, acción, adecuado, anti, antivirus, audio,...

Este grupo parece hablar de seguridad **informática**.

En el café ubicado del lado izquierdo de la Figura 4.5 vemos que dos líneas convergen desde 2006 y 2007 en un tópico del 2009. Observemos primero el tópico final y las dos lineas que llegan.

Tópico café Figura 4.5: Redes.	
2009	transmitir, ancho, banda, cable, energía, aumenta, espera, ieee, adecuado, asignados, basada, busca, canal, capitulo, clase, datos, digital, distancia, estación, ghz, idea, líder, management, mbps,...
2008 linea 1	velocidades, inalámbrico, inalámbrica, mbps, broadcast, cable, inalámbricas, mhz, access, adsl, ancha, antenas banda, cabecera, canal, colisión, compatible, conecta, csma, dedicado,..
2007 linea 1	communications, radiofrecuencia, inalámbrico, inalámbricos, inalámbricas, celulares, wi, adopción, impresora, móvil, rayos, reducida, wireless, acrónimo, alcanzan, añaden, brindado, chips, continuará, cámaras, células, dedico,...
2008 linea 2	ancho, banda, bits, enviados, access, alta, analizar, anteriormente, arquitectura, backbone, cable, capa, conectados, contenido, control, controlador, controladores, crecimiento, cuentan, datos, depende, destino, digital, dinámica, direccionamiento, diseñados, distancia, emplea, enlaces, enviar, espera, extensa, factor, frecuencias, funcionalidad, funcionamiento, gateway, generado, generalmente, interfaces, interferencias, llegar, lógicamente, mac, mbps, modelo, necesarios, network, nodos, obtener, ofrecidos, operaciones, paquete, paquetes, permite, posteriormente, proceso, protocolo, protocolos, proveedor, punto, rango, red, redes, routers, salida, sirven, sistemas, solicitados, surgen, transmiten, transmitir, tráfico, udp, usuarios.
2007 linea 2	ancho, banda, topología, capa, osi, tráfico, enlaces, superiores, wan, activo, asegurar, bits, cabecera, canal, centro, coaxial, combinación, comunicaciones, conectados, conectores, convertir, códigos, detectar, diseñado, distribución, específica, específicas, estrella, fabricantes, fes, frame, inmediata, interior, mbps, máquinas, módem, notificación protocolo, proveedores, pérdida, receptor, recursos, reduce, transfer, transmisiones, transporte, utp.

2006 linea 1	portadora, inalámbricos, trama, celulares, wireless, advanced, congestión, division, etcétera, movilidad, rf, access, ancha, application, gprs, link, móvil, protocol, transparente, tráfico, ul, velocidades.
2007	señalización, isdn, trama, comunicando, congestión, enrutamiento, portadora, analógicas, kilómetros, repetidores, tramas, admite, analógica, conmutación, pulsos, telefonía, telefónicos, transporta.
2006	conmutador, isdn, itu, computadores, gubernamental, opere, asistir, conteo, equipamiento, león, localidad, nula, oro, protege, provenientes, vulnerabilidades.
2006	datagramas, congestión, routing, domain, broadcast, encabezado, extensas, inalámbricas, ipsec, networks, relay, tramas,...

Resulta interesante que la línea 2 pareciera hablar de **redes en general** y la línea 1 está enfocada en **redes inalámbricas**. Se puede analizar también que la linea 1 va desde un año más atrás (2007). Se aprecia que el tema principal es redes inalámbricas pero encontramos que este tópico se divide hacia adelante en otro. Ese tópico a mi parecer está enfocado a **redes analógicas** y está alimentado por otros 2 tópicos del 2006.

Todo este modulo se podría englobar como **redes**, aunque se ve que encuentra distintos subtemas o especialidades y se unen en algunos puntos.

Un ejemplo mas que se puede ver es el tópico Tópico verde claro de la Figura 4.5 en este se puede ver una línea desde 2004 hasta 2008.

Tópico verde claro Figura 4.5: Redes II.	
2004	duplex, full, half, fddi, rate, viajan, atm, congestión, eia, fiber, firmware, gbps, mtu, nic, pbx, retransmisión, trenzado, vlan, wan,...
2005	address, multicast, tramas, broadcast, encaminar, enrutador, fast, gratuita, pasarela, ruteo, trama, adapter, comience, conmutadores, destinos, gateway, ilusión, mode, netscape, temp.
2005	pstn, telephone, switched, telephony, ccitt, conmutación, identification, itu, multiplexing, voice, cobre, conmutada, pcm, union, conmutado, division, equipment, gateway, modulation, pvc, telecommunications, telefonía, transporta, analógica, analógicos, application, broadband, cal, codee, compañías, conector, conmutador, constituir, ...
2006	portadora, transmitiendo, colisión, conmutada, excesiva, trama, transmisiones, velocidades, atenuación, bridges, cabeceras, cableada, colisiones, compartiendo, destinada, fddi, frecuencias, industriales, ipx, movilidad, mueven, multicast, multimodo, relay, repetidor, situados, stp, tramas, transmitido.

2006	isdn, itu, enrutar, ietf, udp, rdsi, ruteadores, atm, destinos, hosts, relay, spx, arp, conmutación, duplex, eléctricas, exchange, fddi, icmp, ring, smtp, trama, trenzado,...
2006	ópticas, fibras, microondas, ranuras, cobre, datagrama, espectro, interconectar, interferencia, movilidad, centrales, coaxial, fijas, isdn, itu, relay, retransmisión, transmisiones, analógica, analógicas, at, barato, cables, celular, circuito, circuitos, cisco, conectado, conmutación, distribuida, divide, eléctricas, emplea, enlazadas, estrictamente,...
2007	duplex, half, transmitiendo, hubs, circuit, configurados, enviará, interconectan, kbps, metropolitana, publicas, topologías.
2008	duplex, full, half, direccionamiento, mode, rate, sequence, authentication, auto, beneficios, bytes, certificación, clear, cm, fast, inalámbricas, mbps, negociación, networking, non, protocol, router, routers, speed, systems, to, use, wan.

Este tópico aparenta estar muy relacionado con el anterior, ambos parecen referirse a **redes**.

El último ejemplo que se presenta en este trabajo se compone de dos tópicos que a mi parecer están muy relacionados. El primero es el tópico rosa de la Figura 4.5 y el segundo el gris de la misma figura.

Tópico rosa Figura 4.5: Agradecimientos.	
2008	mamá, papá, enseñarme, tías, admiro, civil, darme, primos, sobrinos, amiga, compañero, contribuir, primas, razonamiento, suministrar,
2009	agradezco, dios, agradecimientos, enseñarme, amor, enseñanzas, caminar, colaboración, disponer, expertos, implantación, ingeniero, librerías, manipulación, quiero, respeto, up.
2010	agradecimientos, dios, hermanos, amiga, mamá, agradezco, amigos, enseñanzas, mencionan, rodolfo, salud, surge.
2011	agradezco, haberme, agradecimientos, amistad, hermanos, ayudarme, sueños.

Tópico gris Figura 4.5: Agradecimientos II.	
2013	cariño, hermana, ayudarme, haberme, felicidad, compañeros, creer, agradecimientos, amigo, amigos, apoyado, apoyarme, bautista, brindarme, brindaron, carlos, consejos, corazón, declara, descriptivas, especializada, espíritu, financieros, fortalecer, manual, manuales, paciencia, padres, preliminares, primos, productivo, quiero, relevancia, respectivas, satisfacción, sintácticos, verbos, vocabulario.
2014	amor, darme, brindarme, cariño, compañeros, casas, ingeniero, quiero, alma, apoyado, consejos, enfoques, espero, espíritu, experiencias, firewalls, ing, momentos, oportunidad, otorgado, respeto, reyes, vii.
2015	darme brindarme, dios, amigos, amistad, compañeros, permitirme, amor, haberme, apoyarme, ayudarme, cariño, consejos, hermano, mamá, agradecimientos, brindado, experiencias, pasamos, roa,...
2016	amor, paciencia, incondicional, consejos, agradezco, bibliografía, conacyt, creciente, eficiencia, enseñanzas, estancia, guadalupe, hermanos, llena, modelado, organizada, organizado, padres, permitirme, presentes, sistemático.

2017	amor, amistad, darme, brindarme, paciencia, permitirme, alma, amigos, brindado, darles, dedicatoria, gustaría, habilidad, incertidumbre, metas, nube, quiero, respeto.
2018	beca, conacyt, otorgada, optar, agradezco, permitirme, brindado, brindarme, sensores, agradecimientos, among, antenas, celda, dra, literatura, maestro, movilidad, ocurre, paciencia, presentada.

En mi opinión estos dos últimos tópicos son uno solo, el cual pierde la conexión del año 2011-2012 y de 2012-2013 por alguna razón. Aparentemente en el año 2012 no se encontró este tópico o no alcanzó a superar el umbral para encontrar las conexiones. El tópico va relacionado con los **agradecimientos** que se agregan en las tesis y resulta coherente que sea un tópico recurrente en la mayoría de los años.

Capítulo 5: Conclusiones.

La primera conclusión a la que puedo llegar es que la propuesta matemática de usar las funciones hash para calcular *JCC* de forma probabilística resulta en un acercamiento muy costeable en tiempo de cómputo siendo que la corrida del modelo más tardado, *r5-l400*, demoró 9min aproximadamente realizando la corrida entera sobre todos los años. Esto permitió realizar varios experimentos ya que en aproximadamente una hora se pueden hacer los 9 modelos. La corrida del modelo *r2-l200*, demoró apenas 2min, siendo este el mas rápido.

Una segunda cosa que se puede concluir es que el algoritmo es capaz de minar tópicos de forma efectiva y con sentido para nosotros, aun en esta colección de textos extensos. Se encontraron varios tópicos llamativos y que se repiten en el tiempo.

También se puede decir que el método propuesto para conectar los tópicos en el tiempo año con año entrega resultados plausibles de tal forma que se logra identificar cómo varios tópicos se mantienen en el transcurso de los años.

Sobre los tópicos se puede concluir que el tema general de redes es el más popular ya se encuentra presente en varios años consecutivos y en varias ramas con pequeñas diferencias. Es importante mencionar que el algoritmo completo es capaz de identificar las distintas ramas en vez de unirlas todas en una sola. Otro tema que está presente en prácticamente todos los años es el de los agradecimientos.

Las contribuciones importantes de este trabajo son:

- La finalización del desarrollo de un algoritmo para descargar subconjuntos (y en principio la totalidad) de la base de datos de tesis UNAM desde el sistema web.
- La implementación de un algoritmo para minar tópicos usando las bases ya existentes de forma cíclica para un corpus dividido por bloques.
- El desarrollo de un algoritmo para conectar los tópicos en el tiempo usando el concepto de cuentas.
- Ahora existe una base de datos con los tópicos de las tesis de computo usando distintos parámetros de *SWMH*.

Mis observaciones para un futuro trabajo son las siguientes en orden de importancia para mi:

1. Mejorar el preprocesamiento de los datos para evitar el problema de la mala interpretación de algunos caracteres acentuados.
2. Al notar que existen varios módulos poco conectados considero que seria interesante hacer un estudio similar pero en vez de conectar los tópicos de años contiguos solamente hacerlo con todos los años subsecuentes.

3. Realizar la comparación con un estudio usando las tesis completas o divididas en cantidades distintas o incluso patrones mas complicados como podría ser por capítulos.
4. Implementar un esquema probabilístico de conexión de tópicos en el tiempo.

Fin

Capítulo 6: Anexo: Tópicos completos

Tópico azul-claro Figura 4.4: Protocolos de Redes.	
2008	77 protocolo:12 protocolos:11 protocol:8 paquetes:5 dispositivos:4 tcp:4 access:3 request:3 comandos:2 investigar:2 ip:2 recepción:2 transferencia:2 at:1 banda:1 cable:1 cisco:1 clientes:1 comments:1 comprobación:1 comunicarse:1 conexiones:1 conexión:1 configuraciones:1 connection:1 costo:1 data:1 dato:1 desempeño:1 desventajas:1 detallado:1 dirigir:1 diseñada:1 dispositivo:1 dividido:1 entregado:1 enviando:1 especificación:1 ftp:1 hardware:1 identificador:1 informáticos:1 inglés:1 llega:1 lugares:1 lógico:1 mac:1 mensajes:1 máquina:1 necesidad:1 network:1 niveles:1 números:1 packet:1 panorama:1 paquete:1 paralelo:1 permiten:1 proporcionada:1 puerto:1 redes:1 relay:1 router:1 septiembre:1 service:1 servicio:1 servidores:1 señales:1 significado:1 sintaxis:1 soporta:1 soporte:1 system:1 transmisión:1 tráfico:1 área:1
2009	46 tcp:20 ip:17 protocolos:11 protocolo:7 protocol:4 direcciones:3 ftp:3 transferencia:3 abiertos:2 access:2 bloque:2 comunicaciones:2 conexiones:2 enrutamiento:2 espera:2 tráfico:2 administrador:1 asociadas:1 auditoria:1 avance:1 basadas:1 beneficios:1 bit:1 bits:1 byte:1 capas:1 caracteres:1 central:1 comunicarse:1 conectividad:1 consta:1 correcta:1 correo:1 crea:1 data:1 denominada:1 desarrollado:1 dns:1 documentos:1 eficiencia:1 enfoca:1 enviado:1 enviar:1 envío:1 equipo:1 etcétera:1 ethernet:1 evolución:1 extensión:1 falla:1 file:1 flexibilidad:1 físicas:1 global:1 hipertexto:1 host:1 htm:1 identidad:1 imprescindible:1 inglés:1 iso:1 lan:1 lee:1 local:1 lograr:1 líneas:1 mail:1 mayoría:1 mx:1 necesarias:1 osi:1 paquete:1 paquetes:1 privada:1 públicas:1 realmente:1 recibido:1 recibir:1 recurso:1 redundancia:1 respaldo:1 significa:1 traducción:1 transfer:1 transmisión:1 utilice:1 índices:1 única:1
2010	90 puertos:5 tráfico:5 paquetes:4 puerto:4 tcp:3 banda:2 bytes:2 disponibilidad:2 ethernet:2 ftp:2 ip:2 lan:2 protocolos:2 soporta:2 udp:2 vía:2 actuar:1 ancho:1 arranca:1 autenticación:1 baja:1 bits:1 cabecera:1 cableado:1 canal:1 canales:1 capas:1 central:1 cifrado:1 comprobación:1 computadora:1 computadoras:1 comúnmente:1 conectada:1 conectados:1 conectividad:1 confiable:1 configuración:1 correctos:1 dedicado:1 destino:1 direcciones:1 dispositivo:1 dns:1 encargados:1 errores:1 esquema:1 establecido:1 extensión:1 fiabilidad:1 firewalls:1 full:1 físicas:1 hub:1 indispensable:1 local:1 lógicas:1 mac:1 mbps:1 mensajes:1 método:1 múltiples:1 ocurre:1 origen:1 paquete:1 país:1 peticiones:1 políticas:1 receptor:1 recibida:1 reduce:1 relativamente:1 remotos:1 ruta:1 ruteo:1 segmento:1 segmentos:1 servidores:1 sesión:1 subred:1 suma:1 superior:1 superiores:1 terminal:1 transmisión:1 transmite:1 transmitir:1 transporte:1 unidades:1 velocidad:1 área:1 única:1
2011	103 tcp:6 protocol:4 tráfico:4 udp:4 bytes:3 network:3 paquetes:3 router:2 address:1 ataque:1 bsd:1 capaces:1 cobertura:1 comandos:1 compartiendo:1 compatibilidad:1 computadoras:1 conexiones:1 configurado:1 conversión:1 creó:1 enrutadores:1 enrutamiento:1 escala:1 específicos:1 esperando:1 establecer:1 gnu:1 hosts:1 implementaciones:1 ip:1 ipv:1 juan:1 logro:1 líneas:1 man:1 mbps:1 media:1 microsoft:1 multipunto:1 openbsd:1 paquete:1 pcs:1 protocolos:1 round:1 subred:1 terminal:1 unix:1
2012	90 lan:4 receptor:4 router:4 ceros:3 emisor:3 network:3 banda:2 fabricantes:2 gateway:2 ieee:2 interconexión:2 osi:2 access:1 administrar:1 análisis:1 arquitecturas:1 asignación:1 autorizadas:1 certificación:1 cisco:1 comunicarse:1 conectar:1 conexiones:1 conexión:1 contraseñas:1 creciente:1 criterios:1 datagramas:1 definida:1 destino:1 determinada:1 editorial:1 enlace:1 enlaces:1 envío:1 específico:1 estaciones:1 ethernet:1 fabricante:1 física:1 host:1 identificación:1 interconectadas:1 interface:1 interoperabilidad:1 mantener:1 máscara:1 nic:1 norma:1 operativo:1 pc:1 permitiendo:1 protocolos:1 proveedores:1 puertos:1 recomendaciones:1 redes:1 remota:1 requerimientos:1 rfc:1 rutas:1 siguiendo:1 system:1 terminal:1 topologías:1 transmite:1 tráfico:1 udp:1 up:1 virtuales:1 únicas:1 útil:1
2012	56 host:9 protocol:7 configuration:5 ip:5 tcp:5 dynamic:3 hosts:3 puerto:3 docs:2 gigabit:2 paquete:2 products:2 router:2 all:1 alta:1 asignación:1 banda:1 bits:1 cable:1 call:1 cisco:1 computadoras:1 conexiones:1 configuraciones:1 default:1 destino:1 direccionamiento:1 distinción:1 específica:1 fast:1 ftp:1 ieee:1 instalado:1 interconexión:1 iso:1 locales:1 modular:1 network:1 paquetes:1 protocolo:1 proveedor:1 puertos:1 relativamente:1 resolución:1 responsable:1 ruta:1 security:1 segmentos:1 series:1 service:1 subred:1

	transmission:1 virtuales:1 wi:1
2013	128 dns:5 domain:3 ipv:3 cpu:2 dhcp:2 host:2 network:2 operativos:2 protocol:2 remoto:2 servidores:2 tcp:2 comando:1 computadora:1 conectarse:1 configurar:1 gateway:1 hora:1 instalado:1 ip:1 lan:1 llegue:1 máscara:1 nodos:1 pc:1 recibe:1 remota:1 sub:1 virtuales:1
2013	195 ancho:5 banda:5 network:4 lan:2 transmitir:2 andrew:1 basados:1 capa:1 clientes:1 comunicar:1 conectar:1 conexión:1 distintas:1 dns:1 duro:1 eficacia:1 funcionar:1 imagen:1 implementa:1 impresoras:1 intervalo:1 ip:1 longitud:1 método:1 ofrecer:1 pa:1 peticiones:1 port:1 pr:1 prioridad:1 proporciona:1 protocolo:1 protocolos:1 puerto:1 red:1 rápido:1 soporte:1 time:1 transferencia:1 tráfico:1 usada:1 versi:1
2014	103 ancho:7 banda:5 medida:3 conectarse:2 gateway:2 mecanismo:2 números:2 señales:2 tcp:2 telecomunicaciones:2 transmisión:2 tráfico:2 utilización:2 acceder:1 ack:1 actuar:1 arquitectura:1 arquitecturas:1 asignación:1 asociados:1 aspectos:1 audio:1 basa:1 bidireccional:1 brinda:1 bytes:1 básica:1 cable:1 cabo:1 cadena:1 carga:1 cargan:1 clientes:1 compañía:1 comunicaciones:1 comunicación:1 conectividad:1 configuración:1 consecuencia:1 corregir:1 dejar:1 directamente:1 diseñado:1 dispositivo:1 dns:1 ec:1 empresas:1 encargado:1 entorno:1 entran:1 enviado:1 enviados:1 enviar:1 envía:1 envío:1 equipos:1 errores:1 estandarización:1 estructurado:1 estándar:1 extremo:1 fabricantes:1 factor:1 firewall:1 formatos:1 físicas:1 host:1 intercambio:1 interconectados:1 internacional:1 ip:1 itu:1 llegado:1 lograr:1 línea:1 mejora:1 multipunto:1 mundial:1 nombres:1 normas:1 ofrecen:1 ordenadores:1 osi:1 paquete:1 paquetes:1 países:1 poe:1 presencia:1 productos:1 proporcionando:1 protocolo:1 protocolos:1 proveedor:1 puerta:1 puerto:1 recepción:1 receptor:1 recuperación:1 red:1 redes:1 registro:1 router:1 ruteo:1 seguridad:1 seguro:1 servicio:1 servidores:1 señalización:1 simultáneamente:1 telefonía:1 telefónica:1 telefónico:1 tema:1 topología:1 traducción:1 transferencia:1 transmitir:1 transporte:1 udp:1 usada:1 usados:1 viajar:1 voz:1 wan:1 www:1
2015	49 tcp:11 ip:10 paquetes:9 protocolo:9 protocolos:8 mensaje:4 protocol:4 capa:3 comparativa:3 networking:3 banda:2 lan:2 local:2 mac:2 network:2 osi:2 redes:2 transmite:2 voz:2 acceso:1 ampliamente:1 archivos:1 bits:1 computadoras:1 comunicación:1 comunicarse:1 conexiones:1 conexión:1 conmutación:1 conocido:1 consumo:1 continuación:1 creado:1 desventajas:1 determinado:1 direcciones:1 dirigir:1 dispositivo:1 distintas:1 distribuidos:1 distribuir:1 encarga:1 entradas:1 enviar:1 esperar:1 establecido:1 generado:1 gpl:1 hogar:1 host:1 identidad:1 implementar:1 inconsistencias:1 infraestructura:1 inicio:1 lenguaje:1 locales:1 límite:1 mensajes:1 microsoft:1 monitoreo:1 necesita:1 necesitan:1 nodos:1 números:1 oficina:1 operativo:1 paquete:1 países:1 pertenecen:1 ping:1 principal:1 programación:1 promedio:1 proporciona:1 pruebas:1 refiere:1 rfc:1 scripts:1 seguridad:1 seguro:1 separación:1 servicio:1 servidor:1 similar:1 teléfonos:1 transporte:1 técnica:1 udp:1 usada:1 virtuales:1 única:1
2016	55 ip:8 protocolo:6 protocolos:6 comunicaciones:4 adoptar:2 dispositivo:2 dispositivos:2 enlace:2 evolutivo:2 falla:2 fallas:2 frecuentemente:2 orientado:2 programado:2 protocol:2 acceder:1 alberto:1 alejandro:1 amiga:1 amigo:1 anteriormente:1 aumentar:1 basa:1 bucle:1 bus:1 cambiar:1 canal:1 causado:1 cd:1 computadoras:1 comunicar:1 comunicarse:1 conectado:1 conectividad:1 conocido:1 cubrir:1 desarrollados:1 descarga:1 elevado:1 empezando:1 encargado:1 estaciones:1 extendida:1 externos:1 fallos:1 fi:1 figuras:1 group:1 htm:1 incondicional:1 independientemente:1 intentos:1 internas:1 interno:1 lector:1 lograr:1 mediano:1 medios:1 mencionadas:1 mercado:1 muestras:1 mx:1 observa:1 ocurra:1 operativo:1 osi:1 panorama:1 partiendo:1 pc:1 procede:1 proporcionando:1 propuso:1 puertos:1 recibido:1 red:1 redes:1 reloj:1 respectivo:1 router:1 sensores:1 serie:1 suministro:1 superiores:1 tarjeta:1 tcp:1 tomar:1 unidad:1 utilicen:1 velocidades:1

Tópico azul claro Figura 4.5: Criptografía.	
2007	15 descifrar:30 cifra:23 criptográfico:21 encripción:20 secreto:19 criptología:18 destinatario:18 secreta:18 negar:16 cifrar:15 criptografía:15 cifrada:13 griego:10 remitente:9 descifrado:8 atacante:5 convirtiendo:5 criptográficas:5 vulnerabilidades:5 asimétrica:4 encriptado:4 encriptar:4 firmas:4 ocultar:4 phtml:4 protegerse:4 amenazas:3 ataque:3 averiguar:3 bruta:3 escape:3 falsa:3 httpd:3 quince:3 repudio:3 robo:3 tienda:3 ambigüedades:2 ataques:2 auditoria:2 binarios:2 comprobó:2 establezcan:2 estadounidense:2 estima:2 introducirse:2 intruso:2 macintosh:2 norte:2 oculto:2 probando:2 proporción:2 protege:2 provocado:2 símbolos:2 virus:2 ab:1 abstracto:1 acceden:1 aceptada:1 acostumbrados:1 agradecimiento:1 aislar:1 alarmas:1 algorithm:1 alguien:1 alma:1 alteración:1 alteran:1 alterar:1 ambigüedad:1 anota:1 análogo:1 aparece:1 apareciendo:1 armar:1 arte:1 asegurarse:1 atómico:1 autenticidad:1 barra:1 barreras:1 basura:1 binario:1 binary:1 carmen:1 cert:1 certificado:1 cifrado:1 cifrados:1 comercio:1 compañías:1 compra:1 comprador:1 comprobar:1 comprometida:1 conducta:1 conecta:1 confiar:1 confidencialidad:1 configurado:1 consiguiendo:1 conversación:1 convierta:1 corrupción:1 cortar:1 detección:1 directo:1 dotar:1 download:1 edificio:1 ejecutables:1 elaborada:1 electricidad:1 electrónicas:1 emisor:1 encuentro:1 enormemente:1 escalable:1 esquina:1 estaciones:1 evidentemente:1 export:1 facilidades:1 falsos:1 famoso:1 fiabilidad:1 found:1 fuerza:1 garantizan:1 grafos:1 grep:1 guerra:1 gz:1 habituales:1 hacerlos:1 haga:1 icmp:1 impresiones:1 inclusive:1 inconveniente:1 incumplimiento:1 inspiración:1 instalarlo:1 jamás:1 joven:1 lenta:1 limitado:1 llaves:1 llegamos:1 magnéticas:1 malicioso:1 man:1 md:1 millón:1 minuto:1 modificada:1 my:1 negativo:1 obligar:1 olvidar:1 pequeñas:1 pequeños:1 permanecen:1 permitirán:1 plantea:1 prevenir:1 privacidad:1 producida:1 programados:1 provista:1 publicadas:1 publicidad:1 públicas:1 quedar:1 reflejar:1 restringida:1 riesgo:1 romper:1 simétrica:1 simétrico:1 solaris:1 temas:1 teóricamente:1 tiendas:1 tmp:1 transmiten:1 trazo:1 universales:1 validez:1 valiosa:1 viajar:1 visitar:1 vulnerables:1 york:1 zxvf:1
2008	16 cifrada:27 aes:23 pretty:17 ipsec:16 pgp:16 descifrar:14 good:14 cifrar:13 descifrado:13 cipher:12 cifrados:10 encryption:10 privacy:9 rsa:9 cifrado:8 force:8 secreta:8 secure:8 ietf:6 kerberos:6 task:6 túneles:6 xor:6 cifradas:5 túnel:5 autentica:4 autenticado:4 autenticidad:4 cbc:4 certificado:4 cifra:4 ssl:4 vpn:4 atacante:3 authentication:3 certificados:3 des:3 dsa:3 intruso:3 parches:3 spoofing:3 tls:3 criptográficas:2 criptográficos:2 garantizando:2 inseguros:2 intrusiones:2 llave:2 misiones:2 pda:2 pki:2 robusto:2 sas:2 sha:2 smtp:2 vulnerar:2 accedida:1 agencia:1 aleatorios:1 alfa:1 analice:1 apple:1 arpanet:1 atacantes:1 autenticar:1 autorizada:1 azar:1 based:1 binaria:1 bps:1 cableada:1 campo:1 ceros:1 coexistencia:1 compartida:1 conectadas:1 configuradas:1 contraseñas:1 cooperativa:1 daño:1 deberían:1 depender:1 desarrolló:1 descifra:1 difíciles:1 digest:1 dirigidos:1 distribution:1 dns:1 débil:1 eap:1 electrónico:1 elevadas:1 emplearon:1 encaminar:1 encapsulación:1 encapsulado:1 engaño:1 enrutador:1 equivalente:1 espectacular:1 evolucionar:1 exclusiva:1 experimental:1 falla:1 firewall:1 fragmentación:1 frequency:1 gusano:1 hash:1 honestidad:1 implicando:1 institute:1 intercambios:1 interceptado:1 intrusos:1 llaves:1 maliciosos:1 manejan:1 manipulaciones:1 manualmente:1 may:1 md:1 mejores:1 menciona:1 mencionados:1 monitor:1 máquinas:1 mínimas:1 negociación:1 netscape:1 nic:1 oculto:1 opiniones:1 osi:1 pad:1 paridad:1 pcmcia:1 pierden:1 pop:1 portadora:1 ppp:1 prefijos:1 presentaba:1 previo:1 primario:1 privadas:1 procedimientos:1 protecciones:1 protege:1 puedan:1 pública:1 radiofrecuencia:1 remitente:1 repudio:1 rivest:1 ronald:1 salto:1 secreto:1 security:1 seleccionado:1 seleccione:1 septiembre:1 similar:1 society:1 soporta:1 ssid:1 stanford:1 throughput:1 traductores:1 triple:1 troncal:1 udp:1 unicast:1 vale:1 viajar:1 wide:1 wlan:1 world:1
2009	32 criptografía:20 criptográfico:17 cifrar:15 repudio:15 descifrado:9 descifrar:9 concientización:6 asimétrica:5 criptográficas:5 dañar:5 destruida:5 cifrado:4 asimétrico:3 autorizada:3 desastres:3 devolución:3 ids:3 posesión:3 atacantes:2 ataques:2 autenticación:2 autenticidad:2 cifrados:2 contingencia:2 hash:2 incumplimiento:2 simétrica:2 vulnerabilidades:2 abuso:1 académicas:1 accidental:1 activo:1 amenaza:1 amenazas:1 antigua:1 antivirus:1 ataque:1 autorizado:1 aviso:1 cargo:1 causas:1 certificado:1 comprimir:1 comprometidos:1 confidencial:1 confidencialidad:1 confusión:1 consecuencias:1 consideración:1 considerará:1 consumir:1 criptográficos:1 crédito:1 códigos:1 daños:1 debilidad:1 denegación:1 desastre:1 detallada:1 difusión:1 disco:1 división:1 divulgación:1 empírica:1 encargados:1 firewalls:1 firma:1 frecuentemente:1 generen:1 guarden:1 humanas:1 identidad:1 identifica:1 identificados:1 incidente:1 indispensable:1 intrusiones:1 inventarios:1 llave:1 matemáticamente:1 medidas:1 meses:1 mf:1 mitigar:1 otorga:1 peligro:1 perfiles:1 periódicas:1 permitido:1 prestigio:1 privada:1 privilegio:1 proteger:1 publicación:1 realidad:1 recuperada:1 respuesta:1 rige:1 robo:1 robustas:1 router:1 secreta:1 seguro:1 sniffer:1 telecomunicaciones:1

	violación:1 vulnerabilidad:1 vulnerables:1
2010	28 descifrar:20 cifrar:16 cifrados:9 criptoanálisis:7 secreta:6 sha:6 rivest:5 simétrica:5 simétrico:5 rsa:4 aes:3 autógrafa:3 confidencial:3 firmas:3 md:3 simétricos:3 asimétrica:2 autentica:2 cifrado:2 confidencialidad:2 criptografía:2 debilidad:2 encryption:2 firmar:2 hash:2 repudio:2 activos:1 aleatorios:1 algoritmos:1 ataques:1 autenticados:1 compartido:1 compartidos:1 conocidas:1 conocidos:1 descifrado:1 destinatario:1 difundido:1 escribiendo:1 esteganografía:1 firma:1 hackers:1 huella:1 infrastructure:1 inseguro:1 intercambia:1 internacionales:1 key:1 llave:1 ocultar:1 pasivos:1 pista:1 pki:1 previo:1 protege:1 proteger:1 públicamente:1 rc:1 recién:1 remitente:1 simétricas:1 sniffing:1 sustituye:1 válida:1
2010	41 criptografía:9 descifrado:9 ocultar:6 cifrar:5 secreto:5 criptoanálisis:4 descifrar:4 autenticación:3 protege:3 repudio:3 secreta:3 asimétrica:2 autenticidad:2 almacenada:1 ataque:1 autoridad:1 autorizada:1 cert:1 cifra:1 cifrada:1 cifrado:1 esteganografía:1 negar:1 oculta:1 privada:1 propuestos:1
2011	154 secreto:5 descifrar:4 secreta:4 descifrado:3 secretos:3 cifrar:2 hellman:1 llaves:1 romper:1
2012	135 asimétrica:4 receptor:4 cifrar:3 descifrar:3 emisor:3 intercepción:3 simétrica:2 activos:1 api:1 autenticar:1 autorización:1 basan:1 cantidades:1 cargas:1 cifrados:1 confianza:1 costo:1 creada:1 criptografía:1 criptográfico:1 criptográficos:1 definen:1 descifrado:1 destinada:1 detectores:1 directa:1 ecuación:1 entornos:1 esfuerzo:1 exponer:1 expuestos:1 familia:1 firmas:1 fácil:1 fáciles:1 imprescindible:1 instancias:1 interactuar:1 intruso:1 llave:1 matemáticas:1 metas:1 minimizando:1 negocios:1 números:1 orientada:1 propósitos:1 protegen:1 protegida:1 proveen:1 quieren:1 requerida:1 requeridas:1 requieran:1 respaldos:1 robo:1 romper:1 secreta:1 sencillas:1 tema:1 ámbito:1
2013	29 cifrar:21 cifrado:13 rsa:10 secreta:9 descifrado:8 descifrar:7 atacante:5 autentificación:4 cifrada:4 encriptación:4 cifra:3 criptografía:3 criptográfico:3 privacy:3 cifradas:2 descifra:2 inseguro:2 públicas:2 robo:2 segura:2 simétrica:2 acrónimo:1 algoritmo:1 algoritmos:1 aparato:1 ataques:1 autenticación:1 autenticar:1 autorizado:1 cerrar:1 claves:1 conversación:1 cortas:1 curvas:1 denegación:1 determinado:1 digital:1 elípticas:1 emisor:1 enviados:1 impedir:1 implementaciones:1 key:1 letras:1 llaman:1 números:1 obliga:1 primos:1 representado:1 simétricas:1 standard:1 sustitución:1 transmisión:1 wep:1 xor:1

Tópico morado Figura 4.5: Seguridad Informática.	
2009	73 ids:7 nids:6 intrusos:5 ips:5 intrusiones:4 honeypots:3 vulnerabilidades:3 alerta:2 ataque:2 firewalls:2 gusanos:2 intrusión:2 snort:2 trafico:2 accesible:1 alertas:1 arbitraria:1 atacante:1 ataques:1 autorizados:1 continente:1 criptografía:1 detector:1 detectores:1 escaneo:1 firewall:1 interconectadas:1 intrusion:1 malicioso:1 operan:1 repudio:1 tree:1
2009	97 ataques:6 cifrado:5 ataque:3 legítimo:3 protocolo:3 vulnerabilidades:3 atacante:2 auditoría:2 intercepción:2 intruso:2 pasivos:2 aparecen:1 autenticado:1 autorizada:1 autorizados:1 contraseña:1 dispositivo:1 electrónico:1 flash:1 gestión:1 interceptada:1 libremente:1 perspectiva:1 políticas:1 protocolos:1 puertos:1 requisito:1 vulnerabilidad:1
2009	116 nmap:7 intrusion:5 nessus:5 ips:3 atacante:2 nids:2 activamente:1 alerta:1 banderas:1 coincidencia:1 exploits:1 msg:1 subredes:1 syn:1 trafico:1
2009	167 ataques:4 autenticación:4 autorizado:4 autorizada:3 ataque:2 auditoría:2 autorizados:2 denegación:2 intrusiones:2 vulnerabilidades:2 acceder:1 activos:1 anónimo:1 causar:1 clasificación:1 claves:1 confidencial:1 contraseña:1 contraseñas:1 cuentas:1 divulgación:1 encabezado:1 escritura:1 hardware:1 identidad:1 normales:1 números:1 original:1 personales:1 política:1 posibles:1 proteger:1 recursos:1 remoto:1 seguros:1 tráfico:1 virus:1 vulnerables:1
2010	68 repudio:6 autorizada:4 autorizados:4 confidencialidad:4 secreto:4 amenazas:3 auditoría:3 autorizadas:3 ataque:2 autorización:2 desastres:2 políticas:2 proteger:2 amenaza:1 débil:1 hackers:1 intruso:1 nsa:1 política:1 robar:1 united:1 virus:1 vulnerabilidad:1
2010	93 ids:6 detection:5 intrusion:5 intrusos:4 malicioso:3 contraseñas:2 proteger:2 alertas:1 amenazas:1 capaces:1 considerarse:1 dejar:1 disco:1 firewall:1 firmas:1 identidad:1 instante:1 lógica:1 motivos:1 packet:1 seguramente:1 snort:1 subir:1 virus:1
2011	125 antivirus:5 autorizado:4 malicioso:4 vulnerabilidades:4 autenticación:2 confidencialidad:2 filtrado:2 parches:2 prevención:2 security:2 sesión:2 virus:2 activado:1 actualización:1 alerta:1 alumnos:1 amenaza:1 ataque:1 ataques:1 buffer:1 cifrar:1 circuitos:1 cisco:1 compromete:1 contraseñas:1 directorio:1 edificio:1 escaneo:1 gestionar:1 implementa:1 inalámbrica:1 inalámbricos:1 incidentes:1 intrusos:1 junio:1 logs:1 network:1 paquetes:1 países:1 peer:1 proteger:1 proxy:1 responsabilidad:1 router:1 spyware:1 transmitir:1 udp:1 unix:1 violación:1 vulnerabilidad:1 víctimas:1
2012	93 prevención:7 ataques:6 ataque:4 amenazas:2 autorizado:2 confianza:2 firewall:2 informáticos:2 acción:1 adecuado:1 anti:1 antivirus:1 audio:1 cae:1 computadoras:1 confidencial:1 confidencialidad:1 delito:1 detección:1 detectar:1 expuestos:1 hardware:1 línea:1 pasar:1 políticas:1 prevenir:1 proporcionan:1 proteger:1 pérdida:1 transmitida:1 trate:1 video:1 vulnerabilidades:1

Tópico café Figura 4.5: Redes.	
2009	115 transmitir:6 ancho:5 banda:5 cable:3 energía:3 aumenta:2 espera:2 ieee:2 adecuado:1 asignados:1 basada:1 busca:1 canal:1 capitulo:1 clase:1 datos:1 digital:1 distancia:1 estación:1 ghz:1 idea:1 líder:1 management:1 mbps:1 mencionar:1 paquetes:1 presente:1 propuestos:1 protocolo:1 realiza:1 red:1 retardo:1 señal:1 siguen:1 transmite:1 usados:1 usuarios:1 utiliza:1
2008 linea 1	110 velocidades:8 inalámbrico:7 inalámbrica:6 mbps:5 broadcast:2 cable:2 inalámbricas:2 mhz:2 access:1 adsl:1 ancha:1 antenas:1 banda:1 cabecera:1 canal:1 colisión:1 compatible:1 conecta:1 csma:1 dedicado:1 digital:1 diseñar:1 división:1 edificios:1 enlace:1 ethernet:1 fi:1 firewall:1 ftp:1 ghz:1 ideal:1 point:1 potencia:1 propósitos:1 radio:1 retardo:1 roaming:1 rutas:1 separadas:1 ssid:1 telecomunicaciones:1 transmitido:1 unidades:1 usb:1 wep:1 wi:1 wireless:1 wlan:1
2008 linea 2	153 ancho:6 banda:6 bits:5 enviados:2 access:1 alta:1 analizar:1 anteriormente:1 arquitectura:1 backbone:1 cable:1 capa:1 conectados:1 contenido:1 control:1 controlador:1 controladores:1 crecimiento:1 cuentan:1 datos:1 depende:1 destino:1 digital:1 dinámica:1 direccionamiento:1 diseñados:1 distancia:1 emplea:1 enlaces:1 enviar:1 espera:1 extensa:1 factor:1 frecuencias:1 funcionalidad:1 funcionamiento:1 gateway:1 generado:1 generalmente:1 interfaces:1 interferencias:1 llegar:1 lógicamente:1 mac:1 mbps:1 modelo:1 necesarios:1 network:1 nodos:1 obtener:1 ofrecidos:1 operaciones:1 paquete:1 paquetes:1 permite:1 posteriormente:1 proceso:1 protocolo:1 protocolos:1 proveedor:1 punto:1 rango:1 red:1 redes:1 routers:1 salida:1 sirven:1 sistemas:1 solicitados:1 surgen:1 transmiten:1 transmitir:1 tráfico:1 udp:1 usuarios:1
2007 linea 1	44 communications:9 radiofrecuencia:9 inalámbrico:8 inalámbricos:7 inalámbricas:6 celulares:4 wi:3 adopción:2 impresora:2 móvil:2 rayos:2 reducida:2 wireless:2 acrónimo:1 alcanzan:1 añaden:1 brindado:1 chips:1 continuará:1 cámaras:1 células:1 dedico:1 distorsión:1 efectuarse:1 encuesta:1 enormes:1 enumeración:1 esperaba:1 exponen:1 exponer:1 fomentar:1 inalámbrica:1 incapaz:1 infrarrojo:1 iniciativas:1 largas:1 led:1 marcadores:1 mueven:1 múltiple:1 noticias:1 packet:1 pcs:1 pese:1 ponemos:1 portátiles:1 recolección:1 suprimir:1 teléfonos:1 terceras:1 velocidades:1 véase:1
2007 linea 2	70 ancho:9 banda:9 topología:5 capa:3 osi:3 tráfico:3 enlaces:2 superiores:2 wan:2 activo:1 asegurar:1 bits:1 cabecera:1 canal:1 centro:1 coaxial:1 combinación:1 comunicaciones:1 conectados:1 conectores:1 convertir:1 códigos:1 detectar:1 diseñado:1 distribución:1 específica:1 específicas:1 estrella:1 fabricantes:1 fes:1 frame:1 inmediata:1 interior:1 mbps:1 máquinas:1 módem:1 notificación:1 protocolo:1 proveedores:1 pérdida:1 receptor:1 recursos:1 reduce:1 transfer:1 transmisiones:1 transporte:1 utp:1
2006 linea 1	72 portadora:7 inalámbricos:5 trama:4 celulares:3 wireless:3 advanced:2 congestión:2 division:2 etcétera:2 movilidad:2 rf:2 access:1 ancha:1 application:1 gprs:1 link:1 móvil:1 protocol:1 transparente:1 tráfico:1 ul:1 velocidades:1
2006	194 conmutador:5 isdn:4 itu:3 computadores:2 gubernamental:2 opere:2 asistir:1 conteo:1 equipamiento:1 león:1 localidad:1 nula:1 oro:1 protege:1 provenientes:1 vulnerabilidades:1
2006	208 datagramas:5 congestión:4 routing:4 domain:3 broadcast:1 encabezado:1 extensas:1 inalámbricas:1 ipsec:1 networks:1 relay:1 tramas:1
2007	90 señalización:5 isdn:4 trama:4 comunicando:3 congestión:3 enrutamiento:3 portadora:3 analógicas:2 kilómetros:2 repetidores:2 tramas:2 admite:1 analógica:1 conmutación:1 pulsos:1 telefonía:1 telefónicos:1 transporta:1

Tópico verde claro Figura 4.5: Redes II.	
2008	101 duplex:9 full:8 half:7 direccionamiento:3 mode:2 rate:2 sequence:2 authentication:1 auto:1 beneficios:1 bytes:1 certificación:1 clear:1 cm:1 fast:1 inalámbricas:1 mbps:1 negociación:1 networking:1 non:1 protocol:1 router:1 routers:1 speed:1 systems:1 to:1 use:1 wan:1
2007	94 duplex:8 half:7 transmitiendo:5 hubs:4 circuit:2 configurados:1 enviará:1 interconectan:1 kbps:1 metropolitana:1 publicas:1 topologías:1
2006	156 portadora:4 transmitiendo:4 colisión:3 conmutada:2 excesiva:2 trama:2 transmisiones:2 velocidades:2 atenuación:1 bridges:1 cabeceras:1 cableada:1 colisiones:1 compartiendo:1 destinada:1 fddi:1 frecuencias:1 industriales:1 ipx:1 movilidad:1 mueven:1 multicast:1 multimodo:1 relay:1 repetidor:1 situados:1 stp:1 tramas:1 transmitido:1
2005	161 multiplexación:4 bridges:3 hubs:3 csmalcd:2 ebcdic:2 encaminamiento:2 repetidor:2 routers:2 transmisores:2 asynchronous:1 carrier:1 circula:1 computador:1 congestión:1 convenciones:1 encamina:1 fddi:1 gateways:1 half:1 infrarrojos:1 llc:1 megabits:1 perturbación:1 placa:1 sense:1 sitúan:1 token:1 trama:1
2005	162 fddi:3 gigabit:3 ranuras:3 ancha:2 anillo:2 conmutada:2 encaminamiento:2 isdn:2 metropolitana:2 repetidor:2 trenzados:2 bridges:1 conmutación:1 definirse:1 delgado:1 engloba:1 extensa:1 full:1 gateways:1 grueso:1 interconectar:1 multiplexación:1 operan:1 puente:1 puentes:1 rdsi:1 relay:1 sna:1 topologías:1 transmitiendo:1 utp:1
2005	169 address:3 multicast:3 tramas:3 broadcast:2 encaminar:2 enrutador:2 fast:2 gratuita:2 pasarela:2 ruteo:2 trama:2 adapter:1 comience:1 conmutadores:1 destinos:1 gateway:1 ilusión:1 mode:1 netscape:1 temp:1
2005	40 pstn:16 telephone:11 switched:8 telephony:8 ccitt:7 conmutación:6 identification:5 itu:5 multiplexing:4 voice:4 cobre:3 conmutada:3 pcm:3 union:3 conmutado:2 division:2 equipment:2 gateway:2 modulation:2 pvc:2 telecommunications:2 telefonía:2 transporta:2 analógica:1 analógicos:1 application:1 broadband:1 cal:1 codee:1 compañías:1 conector:1 conmutador:1 constituir:1 converger:1 cíclica:1 distingue:1 dividida:1 eléctricas:1 enciende:1 entrantes:1 enviadas:1 exchange:1 fija:1 gsm:1 interactivo:1 isdn:1 juego:1 kbps:1 managed:1 mencionado:1 mueva:1 móvil:1 networks:1 notificaciones:1 parcheo:1 posiciones:1 provider:1 public:1 qos:1 rdsi:1 redundancy:1 relay:1 reporta:1 ring:1 satélite:1 sesiones:1 señalización:1 simplemail:1 switch:1 switches:1 telecom:1 telecomunicación:1 telefónicas:1 telefónicos:1 teléfonos:1 tradicionales:1 tramas:1 transmisor:1 transportes:1 versus:1 viajan:1 voz:1 wan:1 wire:1 ws:1
2006	37 isdn:10 itu:8 enrutar:5 ietf:5 udp:5 rdsi:4 ruteadores:4 atm:3 destinos:3 hosts:3 relay:3 spx:3 arp:2 conmutación:2 duplex:2 eléctricas:2 exchange:2 fddi:2 icmp:2 ring:2 smtp:2 trama:2 trenzado:2 anillo:1 caen:1 circuit:1 coaxial:1 confiable:1 confiables:1 congestión:1 conmutada:1 cubrir:1 datagrama:1 datagramas:1 direccionamiento:1 distribuidos:1 emisor:1 equipamiento:1 extremo:1 fibra:1 fijado:1 firewall:1 instalada:1 ipx:1 iso:1 km:1 lados:1 line:1 llegue:1 man:1 multicast:1 pares:1 periodos:1 proveniente:1 proxy:1 puentes:1 pvc:1 ranura:1 ranuras:1 retardo:1 switches:1 tramas:1 transfer:1 wan:1
2006	39 ópticas:11 fibras:10 microondas:7 ranuras:5 cobre:4 datagrama:3 espectro:3 interconectar:3 interferencia:3 movilidad:3 centrales:2 coaxial:2 fijas:2 isdn:2 itu:2 relay:2 retransmisión:2 transmisiones:2 analógica:1 analógicas:1 at:1 barato:1 cables:1 celular:1 circuito:1 circuitos:1 cisco:1 conectado:1 conmutación:1 distribuida:1 divide:1 eléctricas:1 emplea:1 enlazadas:1 estrictamente:1 incremento:1 ochenta:1 oferta:1 puentes:1 radios:1 redundantes:1 rutas:1 ruteadores:1 satélites:1 significativamente:1 siguen:1 sna:1 sofisticados:1 soportan:1 telefonía:1 videoconferencia:1
2004	59 duplex:14 full:12 half:10 fddi:4 rate:3 viajan:3 atm:2 congestión:2 eia:2 fiber:2 firmware:2 gbps:2 mtu:2 nic:2 pbx:2 retransmisión:2 trenzado:2 vlan:2 wan:2 aal:1 activex:1 address:1 am:1 arpanet:1 aui:1 bgp:1 bridge:1 circula:1 coaxial:1 colisión:1 conectado:1 datagrama:1 defensa:1 difiere:1 fast:1 flow:1 frames:1 hdlc:1 intercambiar:1 isdn:1 llc:1 medium:1 microondas:1 multicast:1 multimodo:1 network:1 núcleo:1 passing:1 pmd:1 potentes:1 protocol:1 rip:1 rmon:1 ruteo:1 segmentación:1 señalización:1 stp:1 subcapa:1 subcapas:1 subred:1 subredes:1 tasa:1 tia:1 transceiver:1 transmission:1 transmiten:1 transmitiendo:1

	twisted:1 videoconferencias:1

Tópico rosa Figura 4.5: Agradecimientos.	
2008	89 mamá:10 papá:9 enseñarme:6 tías:4 admiro:2 civil:2 darme:2 primos:2 sobrinos:2 amiga:1 compañero:1 contribuir:1 primas:1 razonamiento:1 suministrar:1
2009	146 agradezco:5 dios:5 agradecimientos:4 enseñarme:4 amor:2 enseñanzas:2 caminar:1 colaboración:1 disponer:1 expertos:1 implantación:1 ingeniero:1 librerías:1 manipulación:1 quiero:1 respeto:1 up:1
2010	162 agradecimientos:5 dios:5 hermanos:3 amiga:2 mamá:2 agradezco:1 amigos:1 enseñanzas:1 mencionan:1 rodolfo:1 salud:1 surge:1
2011	218 agradezco:5 haberme:4 agradecimientos:3 amistad:3 hermanos:2 ayudarme:1 sueños:1

Tópico gris Figura 4.5: Agradecimientos II.	
2013	71 cariño:9 hermana:7 ayudarme:6 haberme:5 felicidad:4 compañeros:2 creer:2 agradecimientos:1 amigo:1 amigos:1 apoyado:1 apoyarme:1 bautista:1 brindarme:1 brindaron:1 carlos:1 consejos:1 corazón:1 declara:1 descriptivas:1 especializada:1 espíritu:1 financieros:1 fortalecer:1 manual:1 manuales:1 paciencia:1 padres:1 preliminares:1 primos:1 productivo:1 quiero:1 relevancia:1 respectivas:1 satisfacción:1 sintácticos:1 verbos:1 vocabulario:1
2014	70 amor:9 darme:8 brindarme:7 cariño:5 compañeros:3 casas:2 ingeniero:2 quiero:2 alma:1 apoyado:1 consejos:1 enfoques:1 espero:1 espíritu:1 experiencias:1 firewalls:1 ing:1 momentos:1 oportunidad:1 otorgado:1 respeto:1 reyes:1 vii:1
2015	35 darme:21 brindarme:13 dios:10 amigos:8 amistad:7 compañeros:7 permitirme:6 amor:5 haberme:4 apoyarme:3 ayudarme:3 cariño:3 consejos:3 hermano:3 mamá:3 agradecimientos:2 brindado:2 experiencias:2 pasamos:2 roa:2 agradecer:1 apreciar:1 autónoma:1 básico:1 consultada:1 consultar:1 corazón:1 dedicación:1 dedicatorias:1 explicará:1 finalidad:1 fortaleza:1 funcione:1 hermanos:1 historia:1 incondicional:1 laboral:1 menciona:1 mexicana:1 momentos:1 móviles:1 normales:1 ocasiones:1 otorgada:1 paciencia:1 padres:1 pequeña:1 planteamiento:1 precio:1 preguntarse:1 presentación:1 profesional:1 profesionales:1 pública:1 recepción:1 requieran:1 reservada:1 sabiendo:1 seguirá:1 segura:1 significativa:1 sonrisa:1 titulación:1 vayan:1 vender:1 verá:1
2016	105 amor:7 paciencia:6 incondicional:5 consejos:4 agradezco:2 bibliografía:1 conacyt:1 creciente:1 eficiencia:1 enseñanzas:1 estancia:1 guadalupe:1 hermanos:1 llena:1 modelado:1 organizada:1 organizado:1 padres:1 permitirme:1 presentes:1 sistemático:1
2017	158 amor:6 amistad:5 darme:4 brindarme:3 paciencia:3 permitirme:2 alma:1 amigos:1 brindado:1 darles:1 dedicatoria:1 gustaría:1 habilidad:1 incertidumbre:1 metas:1 nube:1 quiero:1 respeto:1
2018	33 beca:13 conacyt:12 otorgada:6 optar:5 agradezco:4 permitirme:3 brindado:2 brindarme:2 sensores:2 agradecimientos:1 among:1 antenas:1 celda:1 dra:1 literatura:1 maestro:1 movilidad:1 ocurre:1 paciencia:1 presentada:1

Capítulo 7: Bibliografía.

1: Internet live stats, "*Twitter Usage Statistics*", , https://www.internetlivestats.com/twitter-statistics/

2: Blei, David M., Andrew Y. Ng, and Michael I. Jordan, "*Latent dirichlet allocation*", 2003

3: Jaegul Choo, Changhyun Lee, Chandan K Reddy, and Haesun Park, "*Utopian:User-driven topic modeling based on interactive nonnegative matrix factorization.*", 2013

4: Mundo biblio, "*Los catálogos de fichas en la Biblioteca Nacional del Perú (década de 1990)*", , https://mundobiblio.wordpress.com/category/historia-de-las-fichas-catalograficas/

5: Fuenes-Pineda, Gibrán y Meza-Ruíz, Ivan Vladimir, "*Sampled Weighted Min-Hashingfor Large-Scale Topic Mining*", 2015

6: Hofmann, T, "*Probabilistic latent semantic indexing*", 1999

7: Salton, Gerard y Michael McGill, "*Introduction to Modern Information Retrieval*", 1986

8: R. Baeza-Yates and B. Ribeiro-Neto, "*Modern Information Retrieval*", 1999

9: S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman., "*Indexing by latent semantic analysis*", 1990

10: Bela A. Frigyik, Amol Kapila, and Maya R. Gupta, "*Introduction to the Dirichlet Distribution and Related*", , https://vannevar.ece.uw.edu/techsite/papers/documents/UWEETR-2010-0006.pdf

11: Chawala, Ravish, "*Topic Modeling with LDA and NMF on the ABC News Headlines dataset*", 2017, https://medium.com/ml2vec/topic-modeling-is-an-unsupervised-learning-approach-to-clustering-documents-to-discover-topics-fdfbf30e27df

12: Simon Funk, "*Netflix Update: Try This at Home*", 2006, https://sifter.org/~simon/journal/20061211.html

13: Xuerui Wang, Andrew McCallum, "*Topics over Time: A Non-MarkovContinuous-Time Model of Topical Trends*", 2006

14: Dirección general de bibliotecas, "*DGB TESIUNAM*", , http://dgb.unam.mx/index.php/catalogos/tesiunam

15: Voutssas, Juan, El sistema TESIUNAM, 1988

16: Biblioteca central, "*Tesis*", , http://bibliotecacentral.unam.mx/tesis.html

17: Jaccard, Paul, " *Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines.*", 1901

18: Schueffel, Patrick; Groeneweg, Nikolaj; Baldegger, Rico, " *The Crypto Encyclopedia: Coins, Tokens and Digital Assets from A to Z*", 2019

19: Chum, Ondřej. James Philbin, Andrew Zisserman, "*Near Duplicate Image Detection:min-Hash and tf-idf Weighting*", 2008

20: Fuentes Pineda, G., Koga, H., Watanabe, T., "*Scalable object discovery: a hash-based approach to clustering co-occurring visual words*", 2011