Universidad Católica del Norte
Departamento de Ingeniería de Sistemas y Computación
Magíster en Ingeniería Informática

Anteproyecto de Tesis:
Information Retrieval and Web Search
Exploring Low Cost Methods for Retrieving Semantically Similar Terms and Documents

30 de junio de 2015

Tesista: Exequiel Fuentes Lettura
Tutor: Dr. Claudio Meneses Villegas

Índice

1. Resumen **1**

2. Formulación de la propuesta **2**
2.1. Introducción . . . 2
2.2. Fundamentos teóricos . . . 6
2.2.1. Conceptos Básicos . . . 6
2.2.2. Recuperación de Información . . . 8
2.2.3. Indexación Semántica Latente . . . 8
2.2.4. Reglas de Asociación . . . 12
2.3. Motivación del trabajo de investigación . . . 14
2.4. Originalidad de la idea de la tesis . . . 15
2.5. Trabajo relacionado . . . 15
2.5.1. Técnicas para medir la satisfacción de los resultados de las búsquedas . . . 17
2.5.2. Técnicas para obtener la estructura semántica entre términos . . . 17
2.6. Resultados esperados . . . 18

3. Hipótesis de Trabajo **20**

4. Objetivos **21**
4.1. Objetivo general . . . 21
4.2. Objetivos específicos . . . 21

5. Metodología **22**
5.1. Análisis del estado del arte . . . 22
5.2. Construir un conjunto de datos de prueba . . . 22
5.3. Evaluación y selección de métodos de Machine Learning . . . 23
5.4. Diseño del método propuesto . . . 23
5.5. Definir criterios para comparar los métodos candidatos . . . 24
5.6. Desarrollar un conjunto de experimentos para comparar el método propuesto con el método LSI . . . 24
5.7. Análisis de resultados . . . 25
5.8. Publicar resultados en medios científicos . . . 25

6. Plan de Trabajo **26**

1. Resumen

El rápido crecimiento de la Web en las dos últimas décadas ha hecho posible el acceso a una gran cantidad de información. La Web consiste de miles de millones de documentos de hipertexto o hipermedios interconectados entre ellos y que son accesibles a través de Internet. Desde su concepción, la Web ha cambiado dramáticamente nuestra forma de buscar y compartir información.

La cantidad de información en la Web es enorme y crece de forma exponencial. La información en la Web es heterogénea, múltiples páginas pueden contener información similar usando palabras completamente diferentes y/o formatos. Por otro lado, mucho del contenido de las páginas es considerado ruido: links de navegación, anuncios comerciales, políticas de privacidad, etc. Para efectos de recuperación de información sólo una parte del contenido es útil.

Debido a la riqueza y a la diversidad de la información en la Web se requiere de técnicas de recuperación de información. La minería de datos permite descubrir información valiosa y/o conocimiento alojado en hipervínculos, páginas de contenido y registros que almacenan las actividades de los usuarios.

Actualmente, los buscadores más populares están basados en el Modelo de Espacio Vectorial y en la coincidencia de términos de la consulta con las palabras indexadas por el sistema de recuperación. Sin embargo, muchos conceptos u objetos pueden ser descritos de diferentes formas debido al contexto y a la forma en como usamos el lenguaje. Por lo tanto, no siempre los resultados obtenidos satisfacen la consulta del usuario.

Existen técnicas de recuperación de información que buscan obtener información relevante contenida en los documentos que no necesariamente implica realizar coincidencias de términos. El método de Indexación Semántica Latente permite obtener información latente contenida en los documentos, la cual puede estar relacionada semánticamente, pero no necesariamente similares lexicológráficamente. El principal inconveniente de esta técnica es el tiempo de ejecución del algoritmo, el cual hace difícil su apliación en una gran colección de documentos como la Web.

El objetivo de esta investigación es proponer un método alternativo al método de Indexación Semántica Latente basado en Machine Learning para la recuperación de información relevante desde la Web. El método propuesto debe aproximar los resultados obtenidos por el método de Indexación Semántica Latente en términos de la relevancia de la información recuperada y a un costo razonable en términos de tiempo de ejecución.

Durante la investigación se analizarán los trabajos relacionados, se realizará la evaluación y selección de técnicas de Machine Learning, diseño del método propuesto, definición de criterios para comparar los métodos candidatos, desarrollo de un conjunto de experimentos para comparar el método propuesto con el método de Indexación Semántica Latente y para finalizar se realizará el análisis de resultados.

2. Formulación de la propuesta

2.1. Introducción

La Web ha impactado cada aspecto de nuestras vidas. La Web consiste de miles de millones de documentos de hipertexto o hipermedios interconectados entre ellos y que son accesibles a través de Internet. Antes de la Web, encontrar información significaba preguntar a un amigo o a un experto o comprar o pedir prestado un libro. Ahora, con la Web esa información está disponible todo el tiempo y sólo basta una conexión a Internet para tener acceso a ella. No sólo es posible encontrar información, también es posible compartir información o conocimiento con otros.

Actualmente, la Web se ha convertido en un importante canal para realizar negocios, es posible comprar casi todo sin la necesidad de ir físicamente a una tienda. Las redes sociales han permitido interconectar a millones de personas alrededor del mundo, permitiendo la comunicación en tiempo real donde es posible expresar puntos de vista y/u opiniones de un tema en particular [28].

La Figura 2.1 y la Tabla 1 grafican el rápido crecimiento de la Web desde 1993 hasta 2014 [7]. Un usuario de Internet se define como un individuo el cual tiene acceso a Internet a través de un computador o un dispositivo móvil.

En 1995, menos del 1 % de la población mundial tenía acceso a Internet. Actualmente, alrededor del 40 % de la población mundial tiene conexión a Internet. El primer billón de usuarios fue alcanzado en 2005. El segundo billón de usuarios fue alcanzado en 2010. Se estima que el tercer billón de usuarios fue alcanzado en 2014.

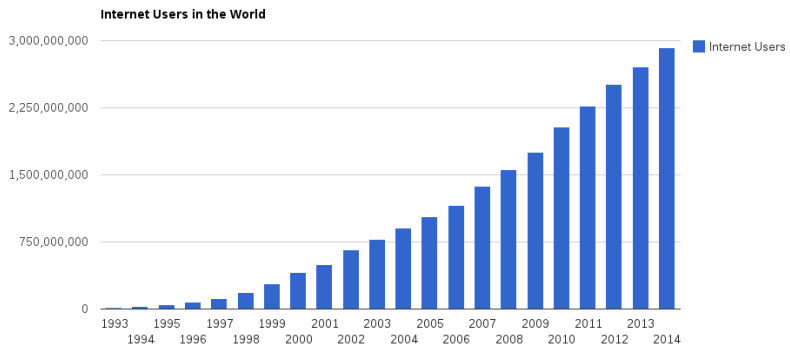

Figura 2.1: Usuarios de Internet en el mundo (fuente: [7])

Los datos de la Tabla 1 fueron actualizados en Julio de 2014 según la fuente [7]. Los datos fueron elaborados por 'International Telecommunication Union (ITU)' y 'United Nations Population Division' [12, 13].

Tabla 1: Porcentaje de población mundial con Internet (fuente: [7])

Year (July 1)	Internet Users	Users Growth	World Population	Population Growth	Penetration (% of Pop. with Internet)
2014*	**2,925,249,355**	7.9%	7,243,784,121	1.14%	**40.4%**
2013	**2,712,239,573**	8.0%	7,162,119,430	1.16%	**37.9%**
2012	**2,511,615,523**	10.5%	7,080,072,420	1.17%	**35.5%**
2011	**2,272,463,038**	11.7%	6,997,998,760	1.18%	**32.5%**
2010	**2,034,259,368**	16.1%	6,916,183,480	1.19%	**29.4%**
2009	**1,752,333,178**	12.2%	6,834,721,930	1.20%	**25.6%**
2008	**1,562,067,594**	13.8%	6,753,649,230	1.21%	**23.1%**
2007	**1,373,040,542**	18.6%	6,673,105,940	1.21%	**20.6%**
2006	**1,157,500,065**	12.4%	6,593,227,980	1.21%	**17.6%**
2005	**1,029,717,906**	13.1%	6,514,094,610	1.22%	**15.8%**
2004	**910,060,180**	16.9%	6,435,705,600	1.22%	**14.1%**
2003	**778,555,680**	17.5%	6,357,991,750	1.23%	**12.2%**
2002	**662,663,600**	32.4%	6,280,853,820	1.24%	**10.6%**
2001	**500,609,240**	21.1%	6,204,147,030	1.25%	**8.1%**
2000	**413,425,190**	47.2%	6,127,700,430	1.26%	**6.7%**
1999	**280,866,670**	49.4%	6,051,478,010	1.27%	**4.6%**
1998	**188,023,930**	55.7%	5,975,303,660	1.30%	**3.1%**
1997	**120,758,310**	56.0%	5,898,688,340	1.33%	**2.0%**
1996	**77,433,860**	72.7%	5,821,016,750	1.38%	**1.3%**
1995	**44,838,900**	76.2%	5,741,822,410	1.43%	**0.8%**
1994	**25,454,590**	79.7%	5,661,086,350	1.47%	**0.4%**
1993	**14,161,570**		5,578,865,110		**0.3%**

Google procesa sobre 40.000 búsquedas en promedio cada segundo, lo cual se traduce sobre 3,5 billones de búsquedas por día [9]. A través de las búsquedas, las personas expresan sus intereses, por ejemplo: qué ver, qué escuchar, qué hacer, qué comprar, qué vender, qué comparar, qué estudiar, etc. Expresar claramente lo que se busca en unas pocas palabras claves es fundamental para encontrar rápidamente lo que se necesita.

Los actuales sistemas de recuperación de información, incluyendo los servicios de búsqueda, tienen una interfaz estándar consistente en una caja de texto que acepta una secuencia de palabras claves. La secuencia de palabras claves es conocida como consulta, la cual es enviada al sistema de recuperación de información, el cual busca coincidencias en una colección de documentos previamente indexados por algún método de ranking, que en la mayoría de los casos es propietario. El sistema de recuperación de información retornará una lista de documentos que posean las mejores coincidencias [18]. La Figura 2.2 muestra la arquitectura general de un sistema de recuperación de información [31].

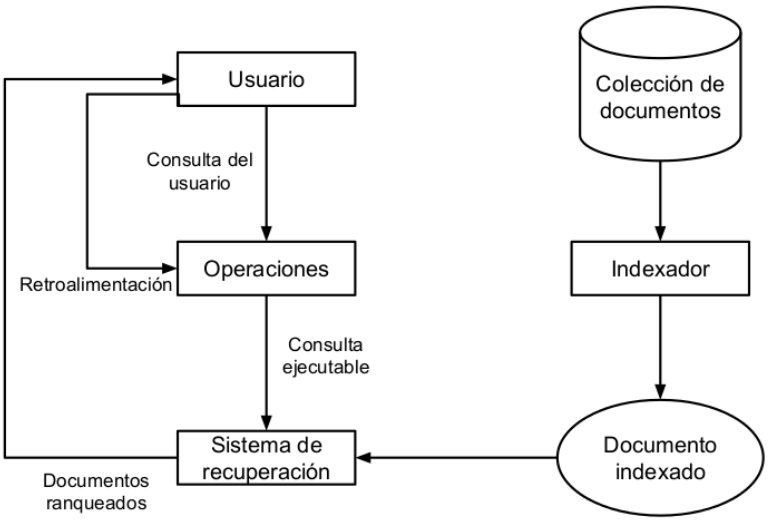

Figura 2.2: Arquitectura General de Recuperación de Información (fuente: [31])

La Figura 2.3 muestra la interfaz de búsqueda de Google. Consiste de un área de texto donde la consulta es ingresada por el usuario denominada 'User Search String'. Los primeros resultados de la lista como la lista al costado derecho se denominan 'Paid Listing', los cuales corresponden a los anuncios publicitarios que coinciden con las palabras claves de la búsqueda. Luego se listan las 10 primeras coincidencias ordenadas por el método de indexación propietario de Google, a esta lista se le denomina 'Organic Listing'.

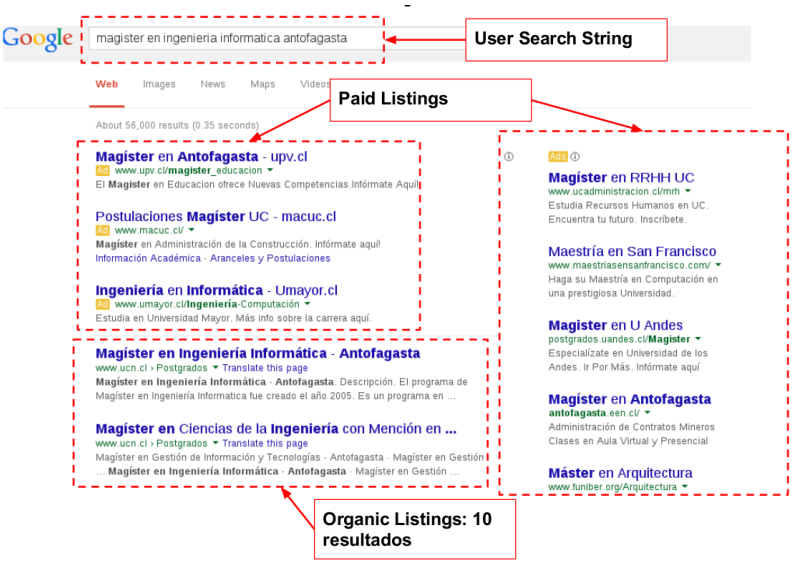

Figura 2.3: Interfaz de búsqueda de Google

Las técnicas de procesamiento de lenguaje natural, minería de datos y Machine Learning trabajan juntos

para automatizar la clasificación y descubrir patrones en los documentos. La principal meta de la minería de datos en documentos de texto es permitir a los usuarios extraer información de recursos de texto y tratar con operaciones como: recuperación, clasificación y análisis de resultados. Eso tiene varios desafíos como: recuperación correcta de los documentos, representación apropiada de los documentos, reducción de la dimensionalidad para manejar los problemas propios de los algoritmos y una apropiada función de clasificación para obtener una buena generalización y evitar el sobreajuste a características muy específicas de los datos de entrenamiento. El procesamiento de lenguaje natural es alcanzar un mejor entendimiento del lenguaje natural por parte de los computadores y representar los documentos semánticamente para incrementar la clasificación y el proceso de recuperación de información. El análisis semántico es el proceso de analizar lingüísticamente las oraciones y las frases en los conceptos claves, los verbos y los sustantivos [15].

Los actuales servicios de búsqueda clasifican las páginas Web para identificar potenciales respuestas a una consulta. El principio comúnmente utilizado para determinar que tan cerca se encuentra un documento (página Web) a un texto (consulta) es calcular la medida de similitud entre ellos, así la mayor probabilidad estimada es la más relevante para el usuario. El conjunto de documentos puede ser visto como un conjunto de vectores en un espacio de vectores, en el cual hay un eje para cada término. La representación común de este conjunto es una matriz $M \times N$ donde las filas representan los M términos (dimensiones) de los N documentos. De manera similar, la consulta es representada como un vector. La similitud del coseno es la medida que consistentemente ha demostrado ser la más efectiva, ya que entrega una puntuación que indica la relación entre el vector consulta y un vector documento. Las puntuaciones resultantes pueden entonces ser utilizadas para seleccionar los documentos que están más relacionados a la consulta [20, 32].

Dado que la consulta del usuario es usualmente corta y escrita en lenguaje natural, el cual es inherentemente ambiguo, los sistemas de recuperación de información cometen errores u omisiones. Entonces, el problema más crítico de recuperación de información es la no coincidencia de los términos utilizados: los indexadores y los usuarios a menudo no utilizan las mismas palabras [18].

Otro enfoque para generar la colección de documentos indexados es utilizar métodos que miden la calidad semántica de los términos. Los términos indexados pueden describir el contexto de los términos en un documento. El método de Indexación Semántica Latente (LSI por sus siglas en inglés) ha demostrado tener resultados competitivos con respecto a métodos estadísticos como TF-IDF (Term Frequency - Inverse Document Frequency). La desventaja de los métodos como LSI es el costo computacional que tienen asociados. Por ejemplo, la complejidad del método LSI es $O(m^2n)$ a diferencia del método TF-IDF cuya complejidad es $O(mn)$, donde m es el número total de documentos en la colección de documentos y n es el número total de términos individuales. Por lo tanto, es complejo aplicar métodos como LSI al contexto de la Web dado el gran volúmen de información existente [40].

El objetivo de esta investigación es proponer un método de Machine Learning para la recuperación parcial o total de información latente en documentos, manteniendo las ventajas del método de Indexación Semántica Latente en cuanto a la relevancia de la información recuperada, pero a un costo computacional menor.

El resto de este documento está organizado como sigue. La sección 2.2 presenta las definiciones teóricas de los métodos utilizados en este documento. La sección 2.3 propone diseñar un nuevo método para resolver el problema de encontrar información latente en documentos. La sección 2.4 refuerza la originalidad de esta investigación. La sección 2.5 presenta el estado del arte, donde se recopila las investigaciones que se han llevado acabo en el contexto de este problema. La sección 2.6 describe que es lo se espera obtener al finalizar este trabajo de investigación. La sección 3 resume la hipótesis sobre la cual será construída esta investigación. La sección 4 presenta el objetivo general y enumera los objetivos específicos. La sección 5 describe las fases que conforman el trabajo de investigación. La sección 6 indica los hitos relevantes y presenta gráficamente

las fases que se ejecutarán durante esta investigación.

2.2. Fundamentos teóricos

Esta sección introduce las técnicas básicas utilizadas en este documento, incluyendo los conceptos de Sistema de Recuperación de Información, Categorización de Texto, Indexación Semántica Latente y Reglas de Asociación[1].

2.2.1. Conceptos Básicos

El conjunto de datos utilizado en la tarea de aprendizaje consiste de un conjunto de registros, el cual puede ser descrito como un conjunto de atributos $A = \{A_1, A_2, ..., A_{|A|}\}$, donde $|A|$ denota el número de atributos contenidos en el conjunto A. Este conjunto también puede contener un atributo especial llamado atributo clase. En la mayoría de la literatura se considera el atributo clase separado del conjunto A. De esta forma, el conjunto de atributos clase C es un conjunto discreto de valores, esto es, $C = \{c_1, c_2, ..., c_{|C|}\}$, donde $|C|$ es el número de clases y $|C| \geq 2$. Un conjunto de datos para aprendizaje es simplemente una tabla relacional, cada registro describe una pieza de una experiencia pasada. En minería de datos un registro también es llamado ejemplo, instancia o caso.

Dado un conjunto de datos D, el objetivo del aprendizaje es producir una función de clasificación/predicción que relacione los valores de los atributos en A con las clases en C. La función puede ser utilizada para predecir el valor de clase de los datos futuros. A esta función se le llama modelo de clasificación, modelo predictivo o simplemente un clasificador.

La tarea de aprendizaje puede ser dividida en dos grandes enfoques: **aprendizaje supervisado** donde las clases son proporcionadas en los datos y **aprendizaje no supervisado** donde las clases son desconocidas y el algoritmo de aprendizaje necesita generar automáticamente los valores de clase.

El conjunto de datos utilizado para el aprendizaje se le denomina **datos de entrenamiento** o **conjunto de entrenamiento**. Después cuando se tenga un modelo entrenado o construído a partir del conjunto de entrenamiento por un algoritmo de entrenamiento, este debe ser evaluado utilizando **datos de prueba** o también conocido como **conjunto de prueba** para determinar la precisión del modelo.

Es importante notar que el conjunto de prueba no es utilizado en el aprendizaje del modelo de clasificación. Los ejemplos en el conjunto de prueba usualmente también poseen valores de clase. Para aprender y probar, los datos disponibles usualmente son divididos en dos subconjuntos, conjunto de entrenamiento y conjunto de prueba. La precisión de un modelo de clasificación en un conjunto de prueba está dado por la Ecuación 1:

$$precisión = \frac{TCC}{TTC} \tag{1}$$

Donde TCC es el número de clasificaciones correctas y TTC es el número total de casos de prueba. Una clasificación correcta significa que el modelo entrenado predijo los mismos valores de clase que los valores de clase originales del ejemplo de prueba. La Figura 2.4 muestra el proceso de aprendizaje. En el paso 1, un algoritmo de aprendizaje utiliza un conjunto de entrenamiento para generar un modelo de clasificación. En el paso 2, el modelo entrenado es probado utilizando el conjunto de prueba para obtener la precisión de la clasificación. Si la precisión del modelo entrenado con el conjunto de prueba es satisfactorio, entonces el modelo puede ser utilizado en las tareas del mundo real para predecir nuevos casos. Si la precisión no es satisfactoria, se necesita volver a comenzar y escoger un algoritmo de aprendizaje diferente y/o realizar adicionalmente un proceso extra sobre los datos (denominado pre-procesamiento de datos). Un proceso de

[1]La sección 'Fundamentos teóricos' se basa en su mayoría de los datos extraídos de [31].

entrenamiento típicamente involucra muchas iteraciones de estos pasos antes de que un modelo entrenado sea satisfactorio. También es posible que no se pueda construir un modelo satisfactorio debido a un alto grado de aleatoriedad en los datos o limitaciones de los algoritmos de aprendizaje.

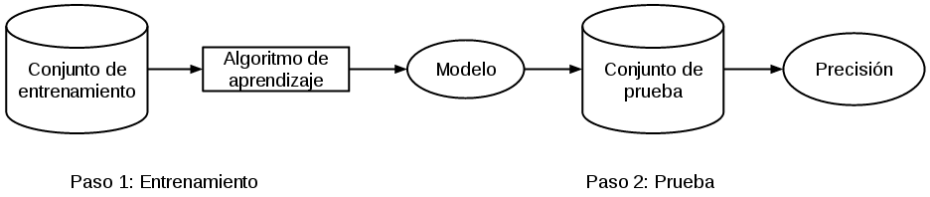

Figura 2.4: Proceso de aprendizaje: entrenamiento y prueba (fuente: [31])

Se ha asumido que el conjunto de datos está disponible, sin embargo, esto no es cierto en la mayoría de los casos. Usualmente, se necesita recolectar los datos crudos, diseñar los atributos y calcular los valores de atributos desde los datos crudos. Los ejemplos más comunes son los datos de textos y aplicaciones Web porque ellos no son compatibles debido a su formato o porque no son correcto o porque no poseen atributos obvios. La categorización de texto esta definida como la asignación de categorias predefinidas a documentos de texto, donde los documentos pueden ser noticias, reportes técnicos, páginas Web, etc. y las categorias a menudo son el tema o también pueden estar basadas en el estilo, pertenencia, etc.

Un documento de texto es considerado una secuencia de oraciones y cada oración consiste de una secuencia de palabras. Un documento es considerado como una bolsa de palabras. La secuencia y la posición de las palabras usualmente es ignorada. Así, un documento puede ser representado como un vector, entonces, dos documentos pueden compararse utilizando la distancia que existe entre dos vectores. La función comúnmente utilizada es la similitud del coseno, la cual es el coseno del ángulo entre el vector de consulta q y el vector del documento d_j

$$cosine(d_j, q) = \frac{\langle d_j \bullet q \rangle}{\|d_j\| \times \|q\|} = \frac{\sum_{i=1}^{|V|} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{|V|} w_{ij}^2} \times \sqrt{\sum_{i=1}^{|V|} w_{iq}^2}} \tag{2}$$

La Figura 2.5 muestra gráficamente como se calcula la similitud del coseno entre el vector de consulta q y los vectores de los documentos d_1, d_2 y d_3. El vector de consulta q y el vector del documento d_j son similares si el ángulo entre ellos es cercano a 0.

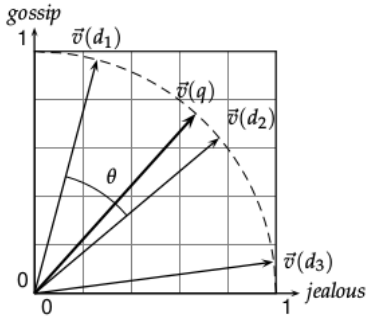

Figura 2.5: Ilustración de la similitud del coseno (fuente: [32])

El ranking de los documentos se calcula utilizando los valores de similitud. Los documentos categorizados con los mayores valores son los más relevantes a la consulta.

2.2.2. Recuperación de Información

La recuperación de información (IR por sus siglas en inglés) es el estudio que ayuda a los usuarios encontrar información que coincida con la información que necesita. Técnicamente, IR estudia la adquisición, organización, almacenamiento, recuperación y distribución de la información.

Un modelo IR guía como un documento y una consulta son representados y que tan relevante es un documento para un usuario. Existen cuatro principales modelos IR: modelo booleano, modelo de espacio vectorial, modelo de lenguaje y modelo probabilístico. Los tres primeros modelos son los comúnmente utilizados en sistemas IR y en la Web.

Aunque estos tres modelos representan los documentos y las consultas de forma diferente, ellos utilizan el mismo framework. Todos ellos tratan cada documento o consulta como una bolsa de palabras o términos. La secuencia de términos y la posición en una oración o en un documento son ignoradas. Un documento es descrito como un conjunto de términos distintos. Un término es simplemente una palabra cuya semántica ayuda a recordar el tema principal del documento. Nótese que el término puede no estar en lenguaje natural en un diccionario. Cada término tiene asociado un peso. Dada una colección de documentos D, $V = \{t_1, t_2, ..., t_{|V|}\}$, un conjunto de términos distintos en la colección, donde t_i es un término. El conjunto V es denominado **vocabulario** de la colección y $|V|$ es el número de términos en V. Un peso $w_{ij} > 0$ está asociado con cada término t_i de un documento $d_j \in D$. Para un término que no aparece en d_j, $w_{ij} = 0$. Cada documento d_j es representado con un vector de términos $d_j = (w_{1j}, w_{2j}, ..., w_{|V|j})$, donde cada peso w_{ij} corresponde al término $t_i \in V$ y cuantifica el nivel de importancia de t_i en el documento d_j. La secuencia de los términos en el vector no es significante. Con esta representación del vector, una colección de documentos es simplemente representada como una tabla relacional (o una matriz). Cada término es un atributo y cada peso es un valor de atributo. Nótese que en diferentes modelos de recuperación, w_{ij} es calculado de forma distinta.

2.2.3. Indexación Semántica Latente

El método de Indexación Semántica Latente (LSI por sus siglas en inglés) fue propuesta por Deewester et al en 1990 [22]. LSI trata de solucionar el problema de determinar la relación semántica de los término con respecto al documento recuperado, identificando una asociación estadística de los términos. Este método asume que hay una estructura semántica latente subyacente en los datos que está parcialmente oculta por aleatoridad de las palabras escojidas. Este método utiliza una técnica estadística denominada Descomposición en Valores Singulares (SVD por sus siglas en inglés) para estimar esta estructura latente y remover el ruido. Los resultados de esta descomposición son descripciones de términos y documentos basados en la estructura semántica latente derivada de SVD. Esta estructura es denominada 'espacio de concepto oculto' en el cual términos y documentos sintácticamente diferentes están asociados semánticamente. Estos términos y documentos transformados en el espacio de concepto son utilizados para recuperación. La consulta también debe ser transformada en el espacio de concepto antes de ejecutar la recuperación.

Sea D una colección de textos, el número de palabras distintas en D es m y el número de documentos en D es n. LSI comienza con una matriz A de términos-documentos $m \times n$. Cada fila de A representa un término y cada columna representa un documento. Así, cada celda de la matriz A denotada por A_{ij} es el número de veces que el término i ocurre en el documento j.

Descomposición en Valores Singulares. Lo que realiza es descomponer la matriz A en el producto de tres matrices:

$$A = U\Sigma V^T \tag{3}$$

Donde U es una matriz $m \times r$ denominado vector singular izquierdo, las columnas de U son vectores ortogonales, es decir $U^T U = I$. V es una matriz $n \times r$ denominada vector singular derecho, las columnas de V son también ortogonales, es decir $V^T V = I$. Σ es una matriz diagonal $r \times r$ denominada matriz de valores singulares, estos valores están ordenados decrecientemente, es decir $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$. m es el número de filas (términos) en A, representando el número de términos. n es el número de columnas en A, representando el número de documentos. r es el rango de A, $r \leq min(m, n)$.

Una importante característica de SVD es que se puede quitar algunas dimensiones que se consideran insignificantes en el espacio de concepto. En el contexto de recuperación de la información, las dimensiones insignificantes puede representar ruido en los datos y debería ser removido. Así, los k valores singulares en Σ son los valores significantes. La matriz aproximada de A es A_k, por lo que también se reduce el tamaño de U, Σ y V quitando las últimas $r - k$ filas y columnas para obtener:

$$A_k = U_k \Sigma_k V_k^T \tag{4}$$

Es decir, se utiliza un triple de largo k para aproximar la matriz término-documento A. El nuevo espacio es denominado espacio de concepto k. La Figura 2.6 muestra la representación esquemática de la matriz original A y la matriz reducida A_k y sus descomposiciones.

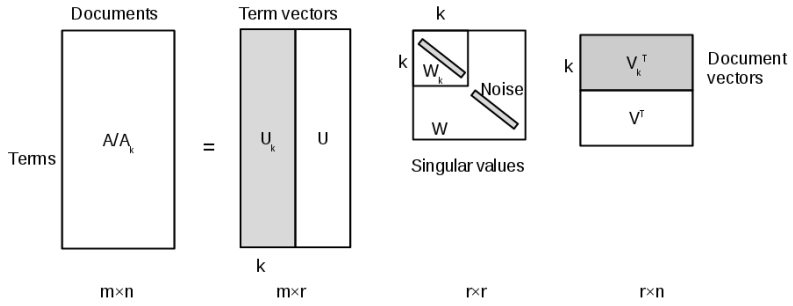

Figura 2.6: Representación esquemática de A y A_k (fuente: [31])

Dado que es una representación reducida, entonces este método no reconstruye la matriz término-documento A perfectamente. Esta representación captura la estructura subyacente más importante con respecto a la asociación entre los términos y los documentos.

Dada una consulta q (representada por una columna en A), primero se convierte a el espacio de concepto k, denotado por q_k. Esta transformación es necesario porque el método SVD ha transformado los documentos originales en el espacio de concepto k y fueron almacenados en V_k. La idea es que q sea tratado como un nuevo documento en el espacio original representado como una columna en A y entonces mapeado en q_k como un documento adicional en V_k^T:

$$q = U_k \Sigma_k q_k^T \tag{5}$$

Dado que las columnas en U son vectores ortogonales, entonces:

$$U_k^T q = \Sigma_k q_k^T \tag{6}$$

Como la inversa de matriz diagonal es aún una matriz diagonal y cada entrada en la diagonal es $1/\sigma_i$ $(1 \leq i \leq k)$, si se multiplica en ambos lados de la ecuación anterior, se obtiene:

$$\Sigma_k^{-1} U_k^T q = q_k^T \tag{7}$$

Finalmente, se obtiene lo siguiente (nótese que la traspuesta de una matriz diagonal es la misma matriz):

$$q_k = q^T U_k \Sigma_k^{-1} \tag{8}$$

Para la tarea de recuperación, simplemente se compara q_k con cada documento en V_k utilizando una medida de similitud, como por ejemplo: la similitud del coseno. Recuerdese que cada fila de V_k (o cada columna de V_k^T) corresponde a un documento en A.

Ejemplo (fuente: [31]): Se tiene una colección de nueve documentos. Los primeros cinco documentos están relacionados con la interacción humano-computador. Los últimos cuatro documentos están relacionados a grafos. Para reducir el tamaño del problema, solamente se utilizarán los términos que están subrrayados.

c_1 : Human machine interface for Lab ABC computer applications
c_2 : A survey of user opinion of computer system response time
c_3 : The EPS user interface management system
c_4 : System and human system engineering testing of EPS
c_5 : Relation of user-perceived response time to error measurement
m_1 : The generation of random, binary, unordered trees
m_2 : The intersection graph of paths in trees
m_3 : Graph minors IV: Widths of trees and well-quasi-ordering
m_4 : Graph minors: A survey

La matriz de término-documento A de 9×12 es la siguiente:

$$A = \begin{array}{c} \begin{array}{ccccccccc} c_1 & c_2 & c_3 & c_4 & c_5 & m_1 & m_2 & m_3 & m_4 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array} \begin{array}{c} \\ human \\ interface \\ computer \\ user \\ system \\ response \\ time \\ EPS \\ survey \\ trees \\ graph \\ minors \end{array}$$

Después de ejecutar SVD, se obtienen tres matrices, U, Σ y V_T. Los valores singulares en la diagonal de Σ están en orden decreciente.

$$U = \begin{pmatrix} 0{,}22 & -0{,}11 & 0{,}29 & -0{,}41 & -0{,}11 & -0{,}34 & 0{,}52 & -0{,}06 & -0{,}41 \\ 0{,}20 & -0{,}07 & 0{,}14 & -0{,}55 & 0{,}28 & 0{,}50 & -0{,}07 & -0{,}01 & -0{,}11 \\ 0{,}24 & 0{,}04 & -0{,}16 & -0{,}59 & -0{,}11 & -0{,}25 & -0{,}30 & 0{,}06 & 0{,}49 \\ 0{,}40 & 0{,}06 & -0{,}34 & 0{,}10 & 0{,}33 & 0{,}38 & 0{,}00 & 0{,}00 & 0{,}01 \\ 0{,}64 & -0{,}17 & 0{,}36 & 0{,}33 & -0{,}16 & -0{,}21 & -0{,}17 & 0{,}03 & 0{,}27 \\ 0{,}27 & 0{,}11 & -0{,}43 & 0{,}07 & 0{,}08 & -0{,}17 & 0{,}28 & -0{,}02 & -0{,}25 \\ 0{,}27 & 0{,}11 & -0{,}43 & 0{,}07 & 0{,}08 & -0{,}17 & 0{,}28 & -0{,}02 & -0{,}05 \\ 0{,}30 & -0{,}14 & 0{,}33 & 0{,}19 & 0{,}11 & 0{,}27 & 0{,}03 & -0{,}02 & -0{,}17 \\ 0{,}21 & 0{,}27 & -0{,}18 & -0{,}03 & -0{,}54 & 0{,}08 & -0{,}47 & -0{,}04 & -0{,}58 \\ 0{,}01 & 0{,}49 & 0{,}23 & 0{,}03 & 0{,}59 & -0{,}39 & -0{,}29 & 0{,}25 & -0{,}23 \\ 0{,}04 & 0{,}62 & 0{,}22 & 0{,}00 & -0{,}07 & 0{,}11 & 0{,}16 & -0{,}68 & 0{,}23 \\ 0{,}03 & 0{,}45 & 0{,}14 & -0{,}01 & -0{,}30 & 0{,}28 & 0{,}34 & 0{,}68 & 0{,}18 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 3{,}34 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2{,}54 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2{,}35 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1{,}64 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1{,}50 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1{,}31 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0{,}85 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0{,}56 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0{,}36 \end{pmatrix}$$

$$V = \begin{pmatrix} 0{,}20 & -0{,}06 & 0{,}11 & -0{,}95 & 0{,}05 & -0{,}08 & 0{,}18 & -0{,}01 & -0{,}06 \\ 0{,}61 & 0{,}17 & -0{,}50 & -0{,}03 & -0{,}21 & -0{,}26 & -0{,}43 & 0{,}05 & 0{,}24 \\ 0{,}46 & -0{,}13 & 0{,}21 & 0{,}04 & 0{,}38 & 0{,}72 & -0{,}24 & 0{,}01 & 0{,}02 \\ 0{,}54 & -0{,}23 & 0{,}57 & 0{,}27 & -0{,}21 & -0{,}37 & 0{,}26 & -0{,}02 & -0{,}08 \\ 0{,}28 & 0{,}11 & -0{,}51 & 0{,}15 & 0{,}33 & 0{,}03 & 0{,}67 & -0{,}06 & -0{,}26 \\ 0{,}00 & 0{,}19 & 0{,}10 & 0{,}02 & 0{,}39 & -0{,}30 & -0{,}34 & 0{,}45 & -0{,}62 \\ 0{,}01 & 0{,}44 & 0{,}19 & 0{,}02 & 0{,}35 & -0{,}21 & -0{,}15 & -0{,}76 & 0{,}02 \\ 0{,}02 & 0{,}62 & 0{,}25 & 0{,}01 & 0{,}15 & 0{,}00 & 0{,}25 & 0{,}45 & 0{,}52 \\ 0{,}08 & 0{,}53 & 0{,}08 & -0{,}03 & -0{,}60 & 0{,}36 & 0{,}04 & -0{,}07 & -0{,}45 \end{pmatrix}$$

Ahora, se escoge solamente los dos valores singulares más grandes de Σ, es decir, $k = 2$. Así, el espacio de concepto tiene sólo dos dimensiones. Las otras dos matrices son también truncadas. Se obtiene las tres matrices U_k, Σ_k y V_k^T.

$$A_k = \begin{pmatrix} 0{,}22 & -0{,}11 \\ 0{,}20 & -0{,}07 \\ 0{,}24 & 0{,}04 \\ 0{,}40 & 0{,}06 \\ 0{,}64 & -0{,}17 \\ 0{,}27 & 0{,}11 \\ 0{,}27 & 0{,}11 \\ 0{,}30 & -0{,}14 \\ 0{,}21 & 0{,}27 \\ 0{,}01 & 0{,}49 \\ 0{,}04 & 0{,}62 \\ 0{,}03 & 0{,}45 \end{pmatrix} \begin{pmatrix} 3{,}34 & 0 \\ 0 & 2{,}54 \end{pmatrix} \begin{pmatrix} 0{,}20 & 0{,}61 & 0{,}46 & 0{,}54 & 0{,}28 & 0{,}00 & 0{,}01 & 0{,}02 & 0{,}08 \\ -0{,}06 & 0{,}17 & -0{,}13 & -0{,}23 & 0{,}11 & 0{,}19 & 0{,}44 & 0{,}62 & 0{,}53 \end{pmatrix}$$

Ahora, se quiere buscar los documentos relevantes a la consulta q 'user interface'. Primero, se debe transformar la consulta q al espacio de concepto k utilizando la Ecuación 8, esto es q_k.

$$q_k = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}^T \begin{pmatrix} 0{,}22 & -0{,}11 \\ 0{,}20 & -0{,}07 \\ 0{,}24 & 0{,}04 \\ 0{,}40 & 0{,}06 \\ 0{,}64 & -0{,}17 \\ 0{,}27 & 0{,}11 \\ 0{,}27 & 0{,}11 \\ 0{,}30 & -0{,}14 \\ 0{,}21 & 0{,}27 \\ 0{,}01 & 0{,}49 \\ 0{,}04 & 0{,}62 \\ 0{,}03 & 0{,}45 \end{pmatrix} \begin{pmatrix} 3{,}34 & 0 \\ 0 & 2{,}54 \end{pmatrix}^{-1} = \begin{pmatrix} 0{,}179 & -0{,}004 \end{pmatrix}$$

Entonces, q_k es comparada con cada vector documento en V_k utilizando la similitud del coseno dado por la Ecuación 2. Los valores son los siguientes:

$$\begin{aligned} c_1 &: \quad 0{,}964 \\ c_2 &: \quad 0{,}957 \\ c_3 &: \quad 0{,}968 \\ c_4 &: \quad 0{,}928 \\ c_5 &: \quad 0{,}922 \\ m_1 &: \quad -0{,}022 \\ m_2 &: \quad 0{,}023 \\ m_3 &: \quad 0{,}010 \\ m_4 &: \quad 0{,}127 \end{aligned}$$

Se obtiene el siguiente ranking final $(c_3, c_1, c_2, c_4, c_5, m_4, m_2, m_3, m_1)$.

2.2.4. Reglas de Asociación

Las reglas de asociación fueron introducidas en 1993 por Agrawal et al [14]. Las reglas de asociación es una de los más importantes modelos para encontrar regularidades en los datos. Es tal vez, el modelo que ha sido

más estudiado por la comunidad de base de datos y minería de datos. Su objetivo es encontrar todas las relaciones de co-ocurrencias llamadas asociaciones.

Este modelo es de hecho muy general y puede ser usado en muchas aplicaciones. Por ejemplo, en el contexto de la Web y en documentos de texto puede ser utilizado para encontrar la relación de co-ocurrencia de palabras y patrones.

El problema de minado con reglas de asociación puede ser escrito como sigue. Dado $I = i_1, i_2, ..., i_m$ un conjunto de 'items'. Dado $T = (t_1, t_2, ..., t_n)$ un conjunto de transacciones, donde cada transacción t_i es un conjunto de 'items' tal que $T_i \subseteq I$. Una regla de asociación es una implicación de la forma:

$$X \to Y, \text{ donde } X \subset I, Y \subset I, \text{ y } X \cap Y = \emptyset$$

Donde X o (Y) es un conjunto de 'items' llamado 'itemset'.

Ejemplo 1: Se quiere analizar cuantos artículos vendidos en un supermercado están relacionados con otros artículos. I es el conjunto de todos los artículos vendidos en el supermercado. Una transacción es simplemente un conjunto de artículos vendidos a un cliente. Por ejemplo, una transacción puede ser:

{Beef, Chicken, Cheese}

Una regla de asociación puede ser:

Beef, Chicken $\to$ Cheese

Donde $\{Beef, Chicken\}$ es X y $\{Cheese\}$ es Y. Por simplicidad, los corchetes a menudo son omitidos en las transacciones y reglas.

Una transacción $t_i \in T$ contiene un 'itemset' X si X es un subconjunto de t_i. El 'contador de soporte' de X en T (denotado como $X.count$) es el número de transacciones en T que contienen X. La fortaleza de una regla es medida por el soporte y la confianza.

Soporte: El soporte de una regla $X \to Y$ es el porcentaje de transacciones en T que contiene $X \cup Y$ y puede ser vista como una estimación de la probabilidad $P(X \cup Y)$. Por lo tanto, el soporte de una regla determina que tan frecuente la regla es aplicable en el conjunto de transacciones T. Dado n el número de transacciones en T. El soporte de una regla $X \to Y$ es la Ecuación 9:

$$soporte = \frac{(X \cup Y).count}{n} \tag{9}$$

El soporte es una medida útil porque si es muy baja, entonces la regla puede ocurrir sólo por casualidad. Aún más, en un ambiente de negocios, una regla que cubra unos pocos casos o transacciones puede no ser útil para obtener una ganancia.

Confianza: La confianza de una regla $X \to Y$ es el porcentaje de transacciones en T que contienen a X y también contienen Y. Puede ser visto como una estimación de la probabilidad condicional $P(Y|X)$. La confianza de una regla viene dado por la Ecuación 10:

$$confianza = \frac{(X \cup Y).count}{X.count} \tag{10}$$

Por lo tanto, la confianza determina la predictibilidad de una regla. Si la confianza de una regla es muy baja, no se puede confiar o inferir o predecir Y dado X. Una regla con baja predictibilidad tiene un uso limitado.

Entonces, dado una transacción T, el problema de minar reglas de asociación es descubrir todas las reglas de asociación en T que posean soporte y confianza mayores o iguales a los valores especificados por el usuario para el soporte mínimo (denotado como minsup) y la confianza mínima (denotada como minconf).

Ejemplo 2: La siguiente lista muestra un conjunto de siete transacciones. Cada transacción t_i es un conjunto de artículos comprados por un cliente. El conjunto I es el conjunto de todos los artículos vendidos en el supermercado.

$$\begin{array}{ll} t_1: & Beef, Chicken, Milk \\ t_2: & Beef, Cheese \\ t_3: & Cheese, Boots \\ t_4: & Beef, Chicken, Cheese \\ t_5: & Beef, Chicken, Clothes, Cheese, Milk \\ t_6: & Chicken, Clothes, Milk \\ t_7: & Chicken, Milk, Clothes \end{array}$$

Dado $minsup = 30\,\%$ y $minconf = 80\,\%$, la siguiente regla de asociación es válida:

$$\text{Chicken, Clothes} \rightarrow \text{Milk [support} = 3/7, \text{confidence} = 3/3]$$

Su soporte es $42.86\,\%$ ($> 30\,\%$) y su confianza es $100\,\%$ ($> 80\,\%$). La siguiente regla también es válida:

$$\text{Clothes} \rightarrow \text{Milk, Chicken [support} = 3/7, \text{confidence} = 3/3]$$

Claramente, más reglas de asociación pueden ser descubiertas. Nótese que la representación de los datos es una simplificación de las compras. En este ejemplo, no se consideró la cantidad ni tampoco el precio.

Un documento de texto o incluso una frase en un sólo documento puede ser tratado como una transacción sin considerar la secuencia de palabras y el número de coocurrencias de cada palabra. Así, dado un conjunto de documentos o conjunto de frases, se puede encontrar relaciones de coocurrencias. Cualquier algoritmo debería ser capaz de encontrar el mismo conjunto de reglas aunque pueden diferir en la eficiencia computacional y los requerimientos de memoria [31].

2.3. Motivación del trabajo de investigación

Como se ha ilustrado en las secciones anteriores, el número de búsquedas en la Web ha crecido constantemente en las última dos decadas. La Tabla 1 y en Figura 2.1 muestran este crecimiento. Se calcula que aproximadamente un $40\,\%$ de la población mundial tiene acceso a Internet.

Los servicios de búsqueda utilizan diversas técnicas para recuperar información de datos procesados previamente. El preprocesamiento de los datos es una parte crucial que influirá en los costos computacionales y en la calidad de la información recuperada. La estrategia fundamental es estudiar la naturaleza de los documentos y su contenido para ser representados en una estructura que permita un rápido acceso. En general, los servicios de búsqueda responden a las consultas de los usuarios haciendo coincidir las palabras claves de la consulta con las palabras indexadas previamente. Debido a las variaciones en la indexación de los documentos y al proceso de búsqueda, los resultados no coinciden entre los servicios de búsqueda. Los objetivos de los servicios de búsqueda es responder a las consultas eficiente y efectivamente. El usuario espera que los resultados obtenidos sean relevantes a su consulta y recuperados rápidamente.

Sin embargo, debido a las características del lenguaje natural es complejo responder inteligentemente a las consultas. Usuarios en diferentes contextos o con diferentes necesidades, conocimiento o hábitos linguísticos

describirán la misma información usando diferentes términos. Así, los resultados obtenidos no necesariamente coinciden con lo que el usuario realmente está buscando. Por ejemplo, 'picture', 'image' y 'photo' son sinónimos en el contexto de cámaras digitales. Si un usuario sólo busca la palabra 'picture', entonces documentos que contengan esa palabra serán recuperados y las palabras 'image' y 'photo' posiblemente no estén contenidas en los documentos recuperados.

Existen algunos métodos para detectar la estructura semántica, como por ejemplo el método LSI [40]. Estos métodos determinan la asociación de los términos en el documento. La desventaja de estos métodos es el costo computacional que tienen asociados. Por ejemplo, la complejidad del método LSI es $O(m^2n)$, donde m es el número de palabras distintas en D, n es el número de documentos en D y D es una colleción de textos. Por lo tanto, se hace difícil su aplicación al contexto de la Web por lo que significaría indexar períodicamente miles de millones de páginas.

2.4. Originalidad de la idea de la tesis

En la sección anterior se mencionó que los métodos para detectar la estructura semántica de los términos en los documentos tienen un alto costo computacional, por lo que su aplicación a problemas en el contexto de la Web no es viable, dado que se requieren respuestas muy rápidas o casi instantaneas.

El objetivo de esta investigación es proponer un método de Machine Learning para la recuperación de información. Por ejemplo, un esquema basado en Reglas de Asociación que recupere parcial o totalmente información latente, pero a un costo computacional significativamente menor.

Al momento de escribir esta propuesta no se encontraron estudios relacionados con esta idea de investigación.

2.5. Trabajo relacionado

La Web es considerada uno de los avances más importantes en las telecomunicaciones después del teléfono. La Web ha estado en constante evolución desde sus inicios. El impacto de la Web en la vida cotidiana sigue siendo un tema de estudio. Se puede encontrar una gran cantidad de investigaciones en el campo de la recuperación de información. La Web carece de una estructura semántica, dificultando para una máquina el entendimiento de la información.

Los actuales servicios de búsqueda categorizan las páginas Web identificando potenciales respuestas a una consulta. Entender la consulta es una de las partes cruciales para recuperar documentos relevantes a la consulta. Los autores en [28] indican que desde el punto de vista de la efectividad, la búsqueda en la Web usualmente consiste de tres componentes básicos: entender la consulta, entender el documento y categorizar documentos. En la Web, dada una consulta, una lista categorizada de enlaces a páginas Web es retornada. La lista de enlaces a páginas Web están en orden descendiente de acuerdo al grado de relevancia a la consulta, esto es el grado de coincidencia de los términos.

Los autores en [25] indican que los usuarios recolectan la información por partes, iterativamente y no toda la información es obtenida con una sola consulta. Debido a la complejidad del lenguaje natural y para incrementar la eficiencia del proceso de recuperación se requiere de una serie de consultas realizando modificaciones a la consulta inicial para obtener la información deseada.

Aunque es trivial obtener páginas arbitrarias de la Web, no es trivial extraer información relevante de esas páginas. El texto de interés está contenido sin duda en la página, pero hay otros componentes que no son relevantes, tales como barras de navegación, cabeceras, pie de páginas, avisos comerciales, etc. Los autores en [27] indican que el preprocesamiento de los datos es una tarea importante y por lo tanto necesita

ser realizada adecuadamente para obtener resultados satisfactorios. La tarea de preprocesamiento incluye remover palabras insignificantes del lenguaje, manejar digitos numéricos, puntuación y manejar html en caso de páginas Web.

Los métodos estadísticos son las soluciones dominantes para la recuperación de información. Estos métodos hacen coincidir la consulta construída en lenguaje natural con los datos almacenados. Los autores en [37] mencionan que la mayoría de las técnicas de recuperación de documentos están basados en análisis estadístico de los términos. El análisis estadístico de los términos frecuentes captura solamente la importancia de los términos dentro de un documento. Los actuales métodos están basados en el Modelo de Espacio Vectorial, el cual representa cada documento como un vector de términos. Esta representación incluye el peso de los términos que usualmente son las frecuencias de estos en el documento. Esta representación se construye en un paso llamado Proceso de Indexación y tiene como resultado un Indexador. Los autores en [19] indican que este proceso está constituído por tres pasos básicos: definición de la fuente de los datos, transformación del contenido y la construcción del indexador.

Los autores en [35] indican que un algoritmo de ranking implementa una función que acepta un conjunto de items y retorna una versión ordenadas de ellos. La función toma en cuenta ciertos parámetros que determinan el orden de los items. La misma colección de items puede ser rankeada por diferentes algoritmos y no necesariamente se obtendría el mismo orden. PageRank [16], HITS [29] y SALSA [30] son los algoritmos de ranking que han tenido mayor relevancia en este último tiempo. Los algoritmo de ranking son diseñados considerando el tamaño de la información, la estructura de los datos, el impacto del tiempo de respuesta y como los resultados son presentados. Cada algoritmo de rankeo implícitamente implementa una estrategia de relevancia. Relevancia es un concepto que simboliza el grado de coincidencia entre la información recuperada y la consulta del usuario. Mientras que para un sistema es dificil saber cual información se necesita, para un usuario juzgar los resultados obtenidos es una tarea simple. Por ejemplo, si un usuario escribe ruby como consulta, este puede estar buscando comprar una joya u obtener información acerca del lenguaje de programación Ruby.

EDIT FROM HERE

De acuerdo a lo indicado en [15] y [40], la representación de texto es un aspecto importante en la clasificación de documentos. Los autores en [40] definen la categorización de texto como la asignación de categorias predefinidas a los documentos de texto donde los documentos pueden ser noticias, reportes técnicos, páginas Web, etc. y las categorias son a menudo el asunto o el tema que puede estar basado en el género, pertenencias, etc.

En la literatura es posible encontrar varios métodos para preprocesar los documentos. Los autores en [31] y [36] entregan una serie de técnicas y algoritmos para representar y clasificar documentos. Los autores en [15] muestran gráficamente como se realiza el proceso de clasificación de documentos, entrega un resumen de las técnicas de selección de características en los documentos y revisa brevemente los algoritmos de machine learning más utilizados en la representación y clasificación de documentos.

Tanto los documentos como las consultas deben ser representadas en el mismo espacio. En [28] (—ya utilizado!!—-), los autores indican que las consultas pueden ser clasificadas en múltiples dimensiones, incluyendo objetivos de la búsqueda, temas, tiempo y localización. Esta clasificación tiene algunos desafios como: (1) consultas demasiado cortas, (2) consultas ambiguas, (3) el significado de consulta puede cambiar dependiendo del tiempo y la localización (4) y la consulta puede contener errores.

Los autores en [18] y en [24] indican que la relativa ineficacia de los sistemas de recuperación de información es

porque la consulta está formada por unas pocas palabras claves. Para lidiar con el problema del vocabulario, varios enfoques han sido propuestos. En [24] proponen una metodología para utilizar los resultados de las búsquedas como una fuente externa de conocimiento. Se envía la consulta al servicio de búsqueda y se asume que los resultados obtenidos son relevantes a la consulta. Clasificar estos resultados, permite clasificar la consulta original con una substancial precisión. En [18], los autores proponen utilizar la técnica de expansión de la consulta, agregando términos con significado similar a la consulta original. Se presenta una versión unificada de un gran número de enfoques relacionados con la expansión de la consulta. Se presenta todos los pasos, revisión de las técnicas más utilizadas, desempeño de la recuperación de la información y posibles líneas de investigación.

2.5.1. Técnicas para medir la satisfacción de los resultados de las búsquedas

En [39] los autores indican que encontrar métricas efectivas para evaluar los sistemas de recuperación de información siempre ha recibido especial atención. Monitorear y predecir la satisfacción del usuario es un aspecto importante, por lo que obtener una buena métrica permitiría incrementar el desempeño del sistema de recuperación de información.

Los autores en [28] (——ya utilizado!!—-) y [39] indican que una forma de evaluar la efectividad de los sistemas de búsquedas, es extraer información de los registros de las búsquedas. Aunque cada investigación muestra diferentes técnicas de extracción y evaluación de los resultados, ambos concluyen que es posible medir estadísticamente la satisfacción del usuario utilizando los registros de las búsquedas. Los autores presentan técnicas y metricas que pueden ser consideradas para esta investigación.

Por otro lado, los autores [26] proponen un método semi-supervisado para medir la satisfacción del usuario. El autor hace una revisión de métodos tradicionales indicando estudios previos entregando una rica literatura y realiza experimentos que serán estudiados y posiblemente aplicados a esta investigación.

En esta investigación se analizarán dos alternativas para medir la satisfacción del usuario. Primero, se analizará si es factible realizar una encuesta a los usuarios para medir su experiencia de búsqueda después de la obtención de los resultados, esta alternativa tiene como defecto el juicio subjetivo de los usuarios. La segunda alternativa, es desarrollar una técnica para medir la información latente en los documentos recuperados, los resultados de esta medición pueden ser comparados con los resultados esperados y por lo tanto obtener valores objetivos los cuales pueden ser cuantificados.

2.5.2. Técnicas para obtener la estructura semántica entre términos

De acuerdo a lo indicado en [34], determinar la similitud semántica entre términos (o expresiones cortas de texto) que tienen el mismo significado, pero no son lexicológráficamente similares es un desafío clave en varios campos de la computación. Muchos de los actuales sistemas de recuperación de información tratan las palabras como unidades atómicas, por lo que no hay noción de similitud entre las palabras.

La clasificación de la consulta por su semántica es útil para la presentación de resultados relevantes [28] (——ya utilizado!!—-). En [36] y [40] se indica que principal debilidad de los actuales modelos de indexación, incluyendo 'inverted index' y 'TF-IDF', estos modelos no entregan un profundo entendimiento de la semántica de los datos, dejando afuera mucha información crítica.

En [33] los autores presentan y evalúan una colección de técnicas desarrolladas para abordar el problema de similitud semántica. El método propuesto determina el grado de similitud entre los términos o expresiones. El método propuesto consiste en medir que tan a menudo dos términos aparecen en la misma consulta, adaptando la definición de la co-ocurrencia de términos para sus propósitos. Como hipótesis los autores

asumen que los usuarios a lo largo de las búsquedas entregan información que puede ser reutilizada para resolver problemas de similitud de términos. Todos los métodos revisados fueron evaluados utilizando benchmark sobre un conjunto de datos.

Los autores en [23] proponen un estudio y un enfoque supervisado para aprender la relación semántica dado un conjunto de datos. El modelo semántico propuesto consiste de estadísticas de co-ocurrencias parametrizadas asociadas con unidades de texto de un gran conjunto de datos. El método es independiente del conjunto de datos y puede ser utilizado en cualquier colección de textos. Se presentan los resultados en un rango extenso de experimentos indicando la efectividad del método propuesto y que tan competitivo es respecto al estado del arte.

En [34] se presenta un diseño para medir las regularidades sintácticas y semánticas en un conjunto de datos. También indica como los tiempos de entrenamiento y la precisión dependen de la dimensionalidad de los vectores de palabras y el tamaño de conjunto de entrenamiento. Así, este artículo entrega varios experimentos que pueden ser útiles para esta investigación.

En [40] se realiza un estudio comparativo de los modelos para representar texto, tales como 'TF*IDF', 'LSI' and 'multi-word'. De los resultados experimentales se demuestra que 'LSI' tiene un mejor desempeño para la clasificación y recuperación de información relacionada semánticamente en los documentos. A pesar del gran poder discriminante que posee el modelo 'LSI', los autores en [40] y [31] destacan la complejidad computacional y la dificultad para aplicarlo en el contexto de la Web. En [17] se analizan diversas técnicas de recuperación de información para un sistema de recomendación. En esta investigación se confirma que las técnicas basadas en SVD, tal como LSI, tienen muy buenos resultados para recuperar información relevante, pero se destaca su alto costo computacional en términos de tiempo.

Los autores en [38] muestran empíricamente la utilización de LSI para identificar asociaciones entre palabras utilizadas en el código fuente de la aplicación 'Philips Healthcare'. Esta aplicación posee un gran número de módulos, el cual contiene cientos de línea de código, entonces no es obvio en que parte del código pueden haber problemas. Un problema que tiene LSI es determinar el número óptimo de dimensiones k, no hay un consenso general para un número óptimo. El artículo original [22] donde se propone el método, sugiere seleccionar entre 50-350 dimensiones. En la práctica, el valor de k necesita ser determinado por prueba y error. Los autores en [38] muestran empíricamente que para ese caso, el valor de k es 100, también entregan el estado del arte asociado a este problema.

De acuerdo a lo indicado en [31], el modelo de reglas de asociación podría ser capaz de aproximar los resultados de 'LSI' y evitar sus inconvenientes. Destaca que las reglas de asociación son muy eficientes en términos computacionales y además son fáciles de comprender. También descata que existe muy poca investigación en esta área. A pesar del tiempo transcurrido desde que el autor señaló esta línea de investigación, no se ha encontrado estudios al respecto.

2.6. Resultados esperados

Al finalizar la investigación se espera obtener los siguientes resultados:

a. Un método basado en Machine Learning que permita recuperar información desde la Web que capture relaciones semánticas entre términos de un documento.

b. Una evaluación objetiva del potencial de algunos métodos de Machine Learning para recuperar información desde la Web.

c. Evaluación empírica del método propuesto y su comparación con resultados obtenidos con el método LSI.

d. Al menos dos artículos aceptados en revistas indexadas.

3. Hipótesis de Trabajo

Esta investigación esta constituída por las siguientes hipótesis:

Hipótesis 1: Un método basado en Machine Learning, por ejemplo: Reglas de Asociación, puede ser utilizado para recuperar información desde la Web.

Hipótesis 2: El método propuesto puede recuperar información que tenga relación semántica con los términos de la consulta.

La Figura 3.7 muestra la arquitectura de Recuperación de Información modificada. Nótese que el método de indexación es reemplazado por el método que se desarrollará en esta propuesta. Por lo tanto, el Sistema de Recuperación de Información será alimentado por una nueva colección de documentos generado por el método propuesto.

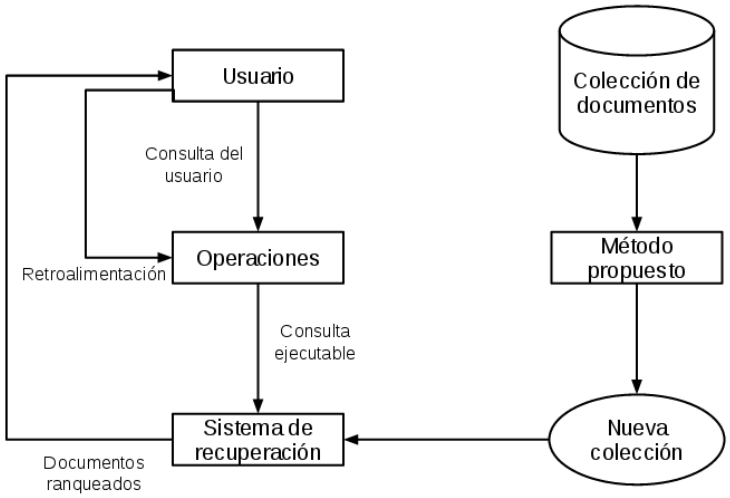

Figura 3.7: Arquitectura propuesta (modificada de fuente: [31])

4. Objetivos

Definida la problemática existente, se plantean el objetivo general y objetivos específicos.

4.1. Objetivo general

Proponer un método basado en Machine Learning para la recuperación de información relevante desde la Web, capturando la estructura semántica subyacente entre los términos y los documentos.

4.2. Objetivos específicos

- Evaluar enfoques de Machine Learning para la recuperación de información de la Web.
- Diseñar un método de Machine Learning para la recuperación de información latente que aproxime los resultados obtenidos por el método de Indexación Semántica Latente, pero a un costo computacional menor.
- Comparar mediante benchmarking el método de Indexación Semántica Latente versus el método de Machine Learning propuesto.
- Difundir los resultados en medios científicos.

5. Metodología

En esta sección se presenta la metodología de trabajo con sus respectivos pasos.

5.1. Análisis del estado del arte

Durante esta fase se busca analizar la información recopilada de fuentes tales como librerías digitales que proporcionan artículos publicados en congresos o revistas relacionados con esta investigación. Esta fase debería prolongarse a lo largo de todo el proceso de desarrollo de esta investigación.

Existen varios congresos y revista que serán exploradas a lo largo de esta investigación. A continuación se enumeran los posibles candidatos:

a. *Journal of the Association for Computing Machinery (JACM).* Es considerada una de las revistas más importante para las Ciencias de Computación. Esta revista cubre una amplia cantidad de investigación de la informática en general y particularmete cubre aspectos teóricos que pueden ser utiles para esta investigación, como por ejemplo: 'Machine Learning and Computational Learning Theory' y 'Web Algorithms and Data Mining' [1].

b. *ACM Digital Library.* Proporciona una colección de documentos de revistas y congresos que han sido financiados por ACM [8].

c. *IEEE Computer Society.* Aunque tiene un enfoque hacia temas relacionados con hardware y estandarización, también proporciona estudios avanzados en 'Machine Learning' y 'Data Mining' [10].

d. *IEEE Xplore.* Similar a *ACM Digital Library*, proporciona una colección de documentos de revistas y congresos que han sido financiados por IEEE [11].

e. *Springer Machine Learning.* Publica artículos que reportan resultados sustanciales en un gran cantidad de métodos de aprendizaje aplicado a una gran variedad de problemas [4].

f. *Springer Data Mining and Knowledge Discovery.* Publica artículos técnicos originales en investigación y aplicacación en el área de 'Data Mining' que son de especial interés para esta investigación [3].

5.2. Construir un conjunto de datos de prueba

El objetivo de esta fase es construir el conjunto de datos de texto que será utilizado para realizar las pruebas de los modelos. Como primera etapa, se realizará una exploración de los conjuntos de datos públicos que están destinados para investigación. Dado que existe cooperación con el grupo 'Center for Semantic Web Research' de la Universidad de Chile [21], es posible que se proporcione un conjunto de datos de texto. Si no es posible obtener un conjunto de datos que cumpla con los requerimientos para propuesta, entonces se diseñará un conjunto de datos de texto. Entonces, las tareas para esta fase son:

a. Exploración de conjunto de datos públicos disponibles.

b. Diseñar un conjunto de datos.

Se tiene una lista preliminar para realizar la exploración de conjuntos de datos públicos, la cual es:

a. *AWS Public Data Sets.* Amazon Web Services almacena una gran variedad de conjunto de datos públicos. Hay conjunto de datos de varios miles de millones de páginas Web denominado 'Common Crawl Corpus'. Para esta investigación, tal vez se requiera crear un subconjunto de este conjunto de datos en caso que este cumpla con los requerimientos para la investigación [2].

b. *UC Irvine Machine Learning Repository.* Al momento de redactar este documento, esta base de datos contenía 307 conjunto de datos. Es necesario hacer una exploración en esta base de datos para determinar si existe algún conjunto de datos candidato para esta investigación [5].

c. *WEKA Data Sets.* Esta base de datos contiene una gran colección de conjuntos de datos los cuales deben ser explorados para determinar si existe algún conjunto de datos candidato para esta investigación [6].

Los autores en [18] describen los pasos para realizar el pre-procesamiento de datos para generar el conjunto de datos. Además, entregan una rica literatura que puede ser considerada para hacer una investigación más profunda en este ámbito.

5.3. Evaluación y selección de métodos de Machine Learning

En esta fase se evaluarán métodos de Machine Learning que permitan extraer información semántica. Cada método será evaluado empíricamente utilizando el conjunto de datos previamente construido. Adicionalmente, cada método será caracterizado en términos de costos computacionales y tipo de información recuperada. Una vez terminadas estas tareas, se seleccionarán los métodos que mejor se ajusten al problema de esta propuesta. Las tareas de esta fase son:

a. Evaluar empíricamente el método.

b. Caracterizar el método, en términos de costos y tipo de información recuperada.

Como se mostró en la sección de 'Fundamentos teóricos', las reglas de asociación permiten descubrir correlaciones entre los términos en un documento, por lo tanto se puede suponer que las reglas de asociación pueden ser capaces de aproximar los resultados de LSI y evitar los costos computacionales asociados. Por lo tanto, este método es un candidato para realizar la evaluación. Cualquier algoritmo debería encontrar el mismo conjunto de reglas de asociación, aunque la eficiencia computacional y requerimientos de memoria puedan ser diferentes. El algoritmo 'Apriori' es uno de los más conocidos, pero existen varias implementaciones que serán analizadas.

En esta etapa es necesario introducir teoría de complejidad computacional para cuantificar la cantidad de recursos que requerirá el método seleccionado. Entonces, se buscará aquel algoritmo más eficiente en término de tiempo y memoria. La eficiencia es esencial para los sistemas de recuperación de información, los resultados deben entregarse lo más rápido posible. Los métodos serán categorizados por clase de complejidad.

Por otro lado, también es fundamental medir la relevancia de los documentos recuperados por el método seleccionado. Para reglas de asociación, es necesario calcular el soporte (9) y la confianza (10). Para otros métodos, será necesario utilizar otras técnicas para medir el desempeño del algortimo. Los autores en [31] y [40] indican que la métrica 'Precision-Recall' permite medir el rendimiento de los sistemas de búsqueda y recuperación de información.

5.4. Diseño del método propuesto

El objetivo de esta fase es diseñar un método que permita recuperar información latente de la Web y a un costo computacional menor a algoritmos tales como Indexación Semantica Latente. Para lograr este objetivo se debe definir un esquema de aprendizaje, definir los parámetros que aceptará el método, sobre que esquema se aplicará y finalmente se debe implementar el método propuesto como un algoritmo. Entonces, las tareas para esta fase son:

a. Definir el esquema de aprendizaje.

b. Definir parámetros del método.

c. Definir esquema de aplicación del modelo inducido.

d. Implementar el método como un algoritmo.

Se enfatiza que el método propuesto debe estar definido en términos formales. El objetivo es construir una función f basado en un método de Machine Learning. Esta función f debe recibir una conjunto de datos D que debe ser aprendido por el algoritmo de aprendizaje. Una vez entrenado el modelo h, se puede ejecutar sobre el conjunto de prueba D' y finalmente medir la capacidad del modelo h para recuperar información latente de la Web.

Similarmente, en esta etapa es necesario utilizar teoría de complejidad computacional para cuantificar la cantidad de recursos que requerirá el nuevo método propuesto. Se debe considerar el espacio en memoria y tiempo de ejecución del algoritmo desarrollado.

Por otro lado, se requerirá medir la relevancia de los documentos recuperados por el nuevo método propuesto.

5.5. Definir criterios para comparar los métodos candidatos

Como se mencionó en las secciones anteriores es necesario construir métricas para cuantificar la cantidad de recursos que requerirán los métodos, así como también desarrollar técnicas para determinar la relevancia de los documentos recuperados por los métodos candidatos. En esta fase se definirán una serie de criterios para realizar la comparación del método propuesto versus los métodos existentes. Deben ser definidos criterios para medir el mejor y el peor caso, dando especial importancia al analisis del peor caso. Por lo tanto, las tareas a realizar en esta fase son:

a. Establecer la forma operativa de medir la relevancia de la información recuperada.

b. Definir la forma de medir los costos computacionales.

5.6. Desarrollar un conjunto de experimentos para comparar el método propuesto con el método LSI

El objetivo de esta fase es desarrollar un conjunto de experimentos para comparar el método propuesto con respecto al método de Indexación Semántica Latente. Estos experimentos permitirán evaluar la efectividad y la eficiencia del nuevo método. En esta etapa es común seleccionar una técnica de benchmarking para realizar la comparación. Una metodología ampliamente utilizada es realizar experimentos sobre diferentes conjuntos de datos, ya sea en tamaño del conjunto de datos para medir la eficiencia, como también ámbitos o temas diferentes para medir la capacidad de generalización del modelo. Los autores en [23] ejecutan varios experimentos que son bastante ejemplificadores para esta investigación, los autores seleccionan diferentes tipos de experimentos y son conducidos paso a paso.

Para lograr este objetivo, se requiere diseñar el experimento a realizar, implementar los métodos en un lenguaje de programación, ejecutar los algoritmos de recuperación de información utilizando el conjunto de datos, medir las variables obtenidas y finalmente resumir los datos. Entonces, las tareas para esta fase son:

a. Diseñar del experimento.

b. Implementar los métodos.

c. Ejecutar algoritmos de recuperación de información.

d. Medir variables del experimento.

e. Resumir datos.

5.7. Análisis de resultados

Después de ejecutar los algoritmos implementados y los algoritmos previamente seleccionados como candidatos para realizar la comparación, se tomarán los datos generados en la fase previa con el objetivo de analizar los resultados obtenidos. Esta etapa incluye el análisis de los costos computacionales, así como también el análisis del desempeño del método propuesto con respecto al método de Indexación Semántica Latente. Entonces, el objetivo de esta etapa es resumir los resultados obtenidos. Así las tareas a desarrollar en esta fase son:

a. Analizar costos computacionales.

b. Analizar desempeño del método propuesto con respecto al método de Indexación Semántica Latente.

5.8. Publicar resultados en medios científicos

Esta fase tiene por objetivo publicar los resultados de la investigación en medios científicos, tales como revistas, congresos o seminarios.

6. Plan de Trabajo

En esta sección se presentan las fases que conforman el plan de trabajo de esta investigación. La Tabla 2 y la Figura 6.8 muestran las fases que se ejecutarán durante la investigación con sus respectivos tiempos de realización. La fase de 'Análisis del estado del arte' se realiza constantemente durante la investigación. Nótese que las fases de 'Diseño del método propuesto' y 'Desarrollo del conjunto de experimentos para comparar el método propuesto con el método LSI' son las que requerirán una mayor cantidad tiempo de desarrollo de la investigación y son consideradas las fases críticas de la investigación. A continuación se enumeran los hitos más relevantes de esta investigación:

Hito 1: Evaluar y seleccionar técnicas de Machine Learning. Cada técnica debe ser evaluada utilizando el conjunto de datos construido para este problema en particular, caracterizada en términos de costos computacionales y tipo de información recuperada. Los resultados obtenidos deben ser resumidos para confeccionar un artículo que será publicado en un congreso. Los pasos a realizar en este hito son:

a. Evaluar los resultados experimentales de cada técnica seleccionada.

b. Publicar en un congreso.

Hito 2: Diseño y prueba del método propuesto. Un nuevo método deber ser diseñado para recuperar información latente de la Web a un costo computacional menor a LSI. El nuevo método deber ser probado con el conjunto de datos construido para este problema en particular, caracterizado en términos de costos computacionales y tipo de información recuperada. Los resultados obtenidos deben ser resumidos para confeccionar un artículo que será publicado en un congreso o revista. Los pasos a realizar en este hito son:

a. Evaluar los resultados experimentales del nuevo método diseñado.

b. Publicar en un congreso o revista.

Hito 3: Análisis de los resultados. Una vez obtenidos los resultados generados en el Hito 1 e Hito 2, se debe proceder a realizar la comparación de los métodos seleccionados versus el nuevo método propuesto. Una tarea fundamental es normalizar los resultados para que estos puedan ser comparados. Se realizará el análisis de los costos computacionales y el análisis del desempeño de los métodos a comparar mediante técnicas de benchmarking. Finalmente, los resultados obtenidos deben ser resumidos para confeccionar un artículo que será publicado en una revista. Los pasos a realizar en este hito son:

a. Comparar métodos seleccionados con el nuevo método propuesto.

b. Analizar desempeño de los métodos.

c. Publicar en una revista.

Tabla 2: Carta Gantt: Tabla

		Name	Duration	Start	Finish	Predecessors
1		Análisis del estado del arte	12.35m	27/11/2014	06/11/2015	
2		Construir un conjunto de datos de prueba	2m	06/03/2015	30/04/2015	
3		Evaluación y selección de técnicas de Machine Learning	2.5m	04/05/2015	10/07/2015	2
4		Diseño del método propuesto	2m	06/07/2015	28/08/2015	2
5		Definir criterios para comparar los métodos candidatos	1m	31/08/2015	25/09/2015	4
6		Desarrollar un conjunto de experimentos para comparar el método propuesto con el método LSI	1m	28/09/2015	23/10/2015	5
7		Análisis de resultados	1m	26/10/2015	20/11/2015	6
8		Publicar resultados en medios científicos	1m	23/11/2015	18/12/2015	7

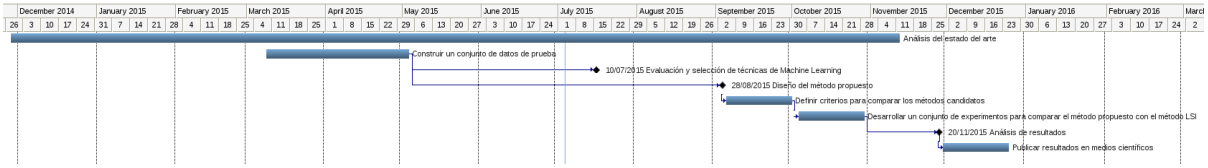

Figura 6.8: Carta Gantt: Gráfico

Referencias

[1] Journal of the association for computing machinery. http://jacm.acm.org/. Online; accessed 20-01-2015.

[2] Public data sets on aws. http://aws.amazon.com/public-data-sets/. Online; accessed 20-01-2015.

[3] Springer data mining and knowledge discovery. http://www.springer.com/computer/database+management+%26+information+retrieval/journal/10618. Online; accessed 20-01-2015.

[4] Springer machine learning. http://www.springer.com/computer/ai/journal/10994. Online; accessed 20-01-2015.

[5] Uc irvine machine learning repository. https://archive.ics.uci.edu/ml/index.html. Online; accessed 20-01-2015.

[6] Weka data sets. http://www.cs.waikato.ac.nz/ml/weka/datasets.html. Online; accessed 20-01-2015.

[7] Internet users. http://www.internetlivestats.com/internet-users/, July 2014. Online; accessed 12-01-2015.

[8] Acm digital library. http://dl.acm.org/, 2015. Online; accessed 20-01-2015.

[9] Google search statistics. http://www.internetlivestats.com/google-search-statistics/, April 2015. Online; accessed 20-04-2015.

[10] Ieee computer society. http://www.computer.org/web/guest, 2015. Online; accessed 20-01-2015.

[11] Ieee xplore. http://ieeexplore.ieee.org/Xplore/home.jsp, 2015. Online; accessed 20-01-2015.

[12] International telecommunication union. http://www.itu.int/en/Pages/default.aspx, 2015. Online; accessed 18-01-2015.

[13] United nations population division. http://www.un.org/en/development/desa/population/, 2015. Online; accessed 18-01-2015.

[14] Agrawal, R., Imieliński, T., and Swami, A. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 1993), SIGMOD '93, ACM, pp. 207–216.

[15] Baharudin, B., Lee, L. H., and Khan, K. A review of machine learning algorithms for text-documents classification. *Journal of Advances in Information Technology 1*, 1 (2010).

[16] Brin, S., and Page, L. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst. 30*, 1-7 (Apr. 1998), 107–117.

[17] Cacheda, F., Carneiro, V., Fernández, D., and Formoso, V. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. *ACM Trans. Web 5*, 1 (Feb. 2011), 2:1–2:33.

[18] Carpineto, C., and Romano, G. A survey of automatic query expansion in information retrieval. *ACM Comput. Surv. 44*, 1 (Jan. 2012), 1:1–1:50.

[19] Ceri, S., Bozzon, A., Brambilla, M., Della Valle, E., Fraternali, P., and Quarteroni, S. The information retrieval process. In *Web Information Retrieval*, Data-Centric Systems and Applications. Springer Berlin Heidelberg, 2013, pp. 13–26.

[20] Das, A., and Jain, A. *Indexing the World Wide Web: The Journey So Far.* 2012, pp. 1–28.

[21] de Chile, U. Center for semantic web research. `http://ciws.cl/`, 2015. Online; accessed 18-01-2015.

[22] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*, 6 (1990), 391–407.

[23] El-Yaniv, R., and Yanay, D. Semantic sort: A supervised approach to personalized semantic relatedness. *CoRR abs/1311.2252* (2013).

[24] Gabrilovich, E., Broder, A., Fontoura, M., Joshi, A., Josifovski, V., Riedel, L., and Zhang, T. Classifying search queries using the web as a source of knowledge. *ACM Trans. Web 3*, 2 (Apr. 2009), 5:1–5:28.

[25] Golitsyna, O., and Maksimov, N. Information retrieval models in the context of retrieval tasks. *Automatic Documentation and Mathematical Linguistics 45*, 1 (2011), 20–32.

[26] Hassan, A. A semi-supervised approach to modeling web search satisfaction. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (New York, NY, USA, 2012), SIGIR '12, ACM, pp. 275–284.

[27] He, Y., Xin, D., Ganti, V., Rajaraman, S., and Shah, N. Crawling deep web entity pages. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining* (New York, NY, USA, 2013), WSDM '13, ACM, pp. 355–364.

[28] Jiang, D., Pei, J., and Li, H. Mining search and browse logs for web search: A survey. *ACM Trans. Intell. Syst. Technol. 4*, 4 (Oct. 2013), 57:1–57:37.

[29] Kleinberg, J. M. Authoritative sources in a hyperlinked environment. *J. ACM 46*, 5 (Sept. 1999), 604–632.

[30] Lempel, R., and Moran, S. Salsa: The stochastic approach for link-structure analysis. *ACM Trans. Inf. Syst. 19*, 2 (Apr. 2001), 131–160.

[31] Liu, B. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*, 2nd ed. Springer Publishing Company, Incorporated, 2011.

[32] Manning, C. D., Raghavan, P., and Schütze, H. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, USA, 2008.

[33] Martinez-Gil, J. An overview of textual semantic similarity measures based on web intelligence. *Artificial Intelligence Review 42*, 4 (2014), 935–943.

[34] Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. *CoRR abs/1301.3781* (2013).

[35] Roa-Valverde, A., and Sicilia, M.-A. A survey of approaches for ranking on the web of data. *Information Retrieval 17*, 4 (2014), 295–325.

[36] Russell, M. A. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More.* O'Reilly Media, Inc., 2013.

[37] Shehata, S., Karray, F., and Kamel, M. An efficient concept-based retrieval model for enhancing text retrieval quality. *Knowledge and Information Systems 35*, 2 (2013), 411–434.

[38] van der Spek, P., and Klusener, S. Applying a dynamic threshold to improve cluster detection of {LSI}. *Science of Computer Programming 76*, 12 (2011), 1261 – 1274. Special Issue on Software Evolution, Adaptability and Variability.

[39] Wang, H., Song, Y., Chang, M.-W., He, X., Hassan, A., and White, R. W. Modeling action-level satisfaction for search task satisfaction prediction. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval* (New York, NY, USA, 2014), SIGIR '14, ACM, pp. 123–132.

[40] Zhang, W., Yoshida, T., and Tang, X. A comparative study of tf*idf, lsi and multi-words for text classification. *Expert Syst. Appl. 38*, 3 (Mar. 2011), 2758–2765.