CENTRO DE CIENCIAS BÁSICAS

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

TESIS

Una Aplicación utilizando CRISP-DM que permita búsquedas optimas de artículos de investigación.

PRESENTA

Edgar Alan Calvillo Moreno

PARA OBTENER EL GRADO DE MAESTRIA EN CIENCIAS DE LA COMPUTACIÓN

COMITÉ DE TESIS

Dr. Alejandro Padilla Díaz

Dr. Julio Cesar Ponce Gallegos

Dr. Jaime Muñoz Arteaga

Aguascalientes, Ags, 15 de Agosto del 2013.

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES

Aguascalientes, Ags, Agosto de 2013

M. en C. José de Jesús Ruíz Gallegos.
Decano del Centro de Ciencias Básicas.
Universidad Autónoma de Aguascalientes.

P R E S E N T E.

Por este conducto hago de su conocimiento que el **L.I. Edgar Alan Calvillo Moreno,** egresado de la Maestría en Ciencias de la Computación del Centro de Ciencias Básicas de la Universidad Autónoma de Aguascalientes, ha integrado satisfactoriamente el documento de tesis titulado: "**Una Aplicación utilizando CRISP-DM que permita búsquedas optimas de artículos de investigación**", por lo que doy mi voto aprobatorio para que continúe con los trámites para presentar el examen de grado reglamentario.

A T E N T A M E N T E

Dr. Alejandro Padilla Díaz
Director de Tesis

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES

Aguascalientes, Ags, Agosto de 2013

M. en C. José de Jesús Ruíz Gallegos.
Decano del Centro de Ciencias Básicas.
Universidad Autónoma de Aguascalientes.

P R E S E N T E.

Por este conducto hago de su conocimiento que el **L.I. Edgar Alan Calvillo Moreno,** egresado de la Maestría en Ciencias de la Computación del Centro de Ciencias Básicas de la Universidad Autónoma de Aguascalientes, ha integrado satisfactoriamente el documento de tesis titulado: "**Una Aplicación utilizando CRISP-DM que permita búsquedas optimas de artículos de investigación**", por lo que doy mi voto aprobatorio para que continúe con los trámites para presentar el examen de grado reglamentario.

A T E N T A M E N T E

Dr. Jaime Muñoz Arteaga
Asesor de Tesis

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES

Aguascalientes, Ags, Agosto de 2013

M. en C. José de Jesús Ruíz Gallegos.
Decano del Centro de Ciencias Básicas.
Universidad Autónoma de Aguascalientes.

P R E S E N T E.

Por este conducto hago de su conocimiento que el **L.I. Edgar Alan Calvillo Moreno,** egresado de la Maestría en Ciencias de la Computación del Centro de Ciencias Básicas de la Universidad Autónoma de Aguascalientes, ha integrado satisfactoriamente el documento de tesis titulado: "**Una Aplicación utilizando CRISP-DM que permita búsquedas optimas de artículos de investigación**", por lo que doy mi voto aprobatorio para que continúe con los trámites para presentar el examen de grado reglamentario.

A T E N T A M E N T E

Dr. Julio Cesar Ponce Gallegos
Asesor de Tesis

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES
Centro de Ciencias Básicas

ANIVERSARIO UAA

L.I. EDGAR ALAN CALVILLO MORENO
ALUMNO (A) DE LA MAESTRÍA EN CIENCIAS CON OPCIÓN A LA COMPUTACIÓN, MATEMÁTICAS APLICADAS
P R E S E N T E .

Estimado (a) alumno (a) Calvillo:

Por medio de este conducto me permito comunicar a Usted que habiendo recibido el voto aprobatorio de su tutor de tesis titulada: "**Una Aplicación utilizando CRISP-DM que permita búsquedas óptimas de artículos de investigación**", hago de su conocimiento que puede imprimir dicho documento y continuar con los trámites para la presentación de su examen de grado.

Sin otro particular me permito saludarle muy afectuosamente.

A T E N T A M E N T E
Aguascalientes, Ags., 20 de septiembre de 2013
"SE LUMEN PROFERRE"
EL DECANO SUSTITUTO

M. en C. JOSÉ DE JESÚS RUIZ GALLEGOS

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES
CENTRO DE CIENCIAS BÁSICAS

c.c.cp.- Interesado
c.c.p.- Secretaría de Investigación y Postgrado
c.c.p.- Jefatura del Depto. de Apoyo al Posgrado
c.c.p.- Consejero Académico
c.c.p.- Minuta Secretario Técnico

JJRG,mjda

Índice General

ÌNDICE GENERAL 1
ÌNDICE DE TABLAS 2
ÌNDICE DE FIGURAS 3
RESUMEN 4
INTRODUCCIÒN 6
CAPÍTULO 1 MÉTODO DE INVESTIGACIÓN 9
1.1. DESCRIPCIÓN DEL CONTEXTO DE LA SITUACIÓN PROBLEMÁTICA DE INVESTIGACIÓN. 9
1.2 RELEVANCIA Y JUSTIFICACIÓN DE LA INVESTIGACIÓN. 10
1.3 DESCRIPCIÓN GENERAL DEL ENFOQUE DE INVESTIGACIÓN. 11
1.4 PROBLEMA DE INVESTIGACIÓN. 12
1.5 PREGUNTA Y OBJETIVO DE INVESTIGACIÓN. 13
1.6 HIPÓTESIS O PROPOSICIONES DE INVESTIGACIÓN. 14
CAPÍTULO 2 MARCO TEÓRICO 15
2.1. DESCUBRIMIENTO DEL CONOCIMIENTO. 15
2.2. CLUSTERIZACION. 16
2.3 MINERÍA DE TEXTOS 22
2.4 MINERÍA DE TEXTOS 23
2.5. MODELO DE INVESTIGACION. 35
2.6DESCRIPCION DE CONSTRUCTOS, VARIABLES OPERACIONALES Y ESCALAS. 43
2.7 DESCRIPCION DE LOS PRINCIPALES ESTUDIOS RELACIONADOS. 43
2.8 ANALISIS DE CONTRIBUCIONES Y LIMITACIONES DE LOS PRINCIPALES ESTUDIOS RELACIONADOS. 43
CAPÍTULO 3 MARCO REFERENCIAL. 46
3.1MODULO DE BASE DE CONOCIMIENTO. 46
3.2MODULO DE CLASIFICACIÓN. 50
3.3BASE DE CONOCIMIENTO UTILIZANDO BAYES. 51
3.4MODULO DE BUSQUEDA. 54
3.5 CLASIFICADOR UTILIZANDO UN WRAPPER 54
3.6 IMPLEMENTACIÓN DE CLASIFICADOR BAYESIANO 55
3.7 IMPLEMENTACIÓN DE WRAPPER 57
3.8 RESULTADOS 59
3.9 ARTÍCULOS PUBLICADOS 63
CAPÍTULO 4 RESULTADOS 66
CONCLUSIONES Y TRABAJOS FUTUROS 68
GLOSARIO DE TÉRMINOS. 73
REFERENCIAS. 75
ANEXOS. 79

Índice de Tablas

TABLA 1. TRABAJOS RELACIONADOS. 45
TABLA 2. ALGUNAS DE LAS PALABRAS UTILIZADAS POR EL CLASIFICADOR BAYESIANO. 50
TABLA 3. PROMEDIO POR FRASES. 51
TABLA 4. OBTENCIÓN DE VALORES POR FRASE SEGÚN LA CATEGORÍA. 52
TABLA 5. VALORES FINALES PARA EL CLASIFICADOR BAYESIANO. 53
TABLA 6. COMPROBACIÓN DEL CLASIFICADOR BAYESIANO “LEAVE-ONE-OUT”. 54
TABLA 7. ACIERTOS DE WRAPPER. 61
TABLA 8. ACIERTOS DE BAYESIANO. 61
TABLA 9. ERRORES EN WRAPPER. 62
TABLA 10. ERRORES EN CLASIFICADOR BAYESIANO. 63
TABLA 11. PUBLICACIONES. 64

Índice de Figuras

FIGURA 1.MODELO DE MINERÍA DE DATOS 16
FIGURA 2. FRAMEWORK UTILIZADO PARA MINERÍA DE TEXTOS 29
FIGURA 3. FASES EN LA METODOLOGÍA CRISP-DM 36
FIGURA 4. METODOLOGÍA PARA LA CLASIFICACIÓN Y BÚSQUEDA DE ARTÍCULOS DE INVESTIGACIÓN SEGÚN CRISP-DM 38
FIGURA 5. ARQUITECTURA DE BÚSQUEDA 40
FIGURA 6. PROCESO PARA DAR DE ALTA UN ARTÍCULO 41
FIGURE 7.1INTERFACE DE BÚSQUEDA PARA EL WRAPPER Y EL CLASIFICADOR BAYESIANO 60
FIGURE 7.2.RESULTADOS DE LA BUSQUEDA 60

RESUMEN

En este trabajo de tesis se investigaron los métodos para identificar patrones en el texto introducido previo a una búsqueda web[1], este procedimiento implementa un procedimiento que funcione como un agente que reconozca los patrones introducidos apoyándose en una base de conocimientos previa que contienen los posibles casos en el área de lenguajes de programación, así como una técnica de búsqueda en modo de clusters[2] para tener un mejor tiempo de respuesta en las búsquedas.

La búsqueda de información por parte de investigadores para la generación de nuevos contenidos lleva a usar motores de búsqueda convencionales para localizar información, estos motores no tienen ningún filtro previo que ayude al investigador a obtener, existen actualmente metabuscadores pero están hechos solo para información que se encuentra previamente procesada para ser localizada por los metabuscadores. Este tipo de limitaciones genera una necesidad para encontrar un método que nos ayude a la clasificación y localización de información contenida en artículos de investigación comúnmente localizada en lenguaje natural, por lo cual se necesita de una herramienta que nos ayude a facilitar el trabajo de los investigadores en la localización de artículos de investigación.

La utilización de técnicas para la clasificación de información lleva como fin la implementación de técnicas comúnmente utilizadas en minería de datos[3] para adaptarlas hacia un funcionamiento asociando directamente documentos de texto mediante técnicas utilizadas en minería de textos donde destaca la utilización de filters que es la utilización de un filtro que ayuda a catalogar la información contenida de un documento utilizando como base principal una base de conocimiento que ayudara a determinar el contenido de dicho contenido mediante frases o palabras que estén asociadas a un tema específico a catalogar, este enfoque es utilizando un esquema de aprendizaje semiautomático debido a que depende en gran parte de la información contenida en la base de conocimiento para lograr una clasificación acertada.

Existen también técnicas de aprendizaje automático[4] que se adaptan al mismo documento y en base a la información que encuentran determinan la clasificación a la cual pertenecen , este tipo de técnicas son llamadas wrappers tienen como desventaja ante un filter el hecho de que tarda más en procesar y clasificar la información debido a que se adaptan al tipo de documento, es decir no necesitan de una base de conocimiento y también es compleja su implementación debido a que se necesita de una meta heurística o heurística para optimizar un poco el funcionamiento, los resultados que se obtienen mediante esta técnica son mucho mejores que los del uso de un filter pero tienen como desventaja el tiempo que tardan en procesar y clasificar el documento.

El presente trabajo de tesis plantea la utilización como base un filter apoyándose en una base de conocimientos que leerá un documento pdf para determinar el tipo de contenido que tiene y clasificarlo según el tipo de información que tenga con las categorías principales de programación, base de datos y sistemas operativos. El uso de una base de conocimiento será la principal herramienta que utilizaremos para clasificar pero tiene como agregado un algoritmo

de colonia de hormigas que ayudara a recorrer el documento pdf de una mejor manera para optimizar el funcionamiento y obtener el mejor resultado posible de la clasificación. Una vez establecida la categoría se asignara el documento al cluster que le corresponde, los clusters serán especificados mediante el algoritmo k-means[5].

INTRODUCCION

La utilización de artículos científicos como medio de comunicación para dar a conocer sus investigaciones es divulgada a través de publicaciones impresas, repositorios de revistas científicas, páginas de los propios autores de artículos o conferencias. Estos documentos son escritos por diferentes investigadores por lo cual cada evento donde es publicado el artículo debe cumplir con un formato predeterminado de la misma manera el medio donde es publicado el artículo es el encargado de buscar la divulgación del trabajo aceptado, estos lugares donde es publicado el trabajo se encarga de lanzar convocatorias de manera frecuente para que el conocimiento sea compuesto con información reciente y buscando un incremento constante en la calidad de los trabajos. Esto es debido al tipo de contenido que se presenta el cual sigue evolucionando con el tiempo acorde a las necesidades del problema que se esté atacando[6].

La productividad de los investigadores, el creciente número de eventos donde se puede presentar o publicar el artículo, los diferentes formatos en los que se envía el documento, el contenido mixto que puede ir desde imágenes, números, formulas y texto ha generado la necesidad de sistemas que permitan la administración de estos contenidos siendo clave la búsqueda y clasificación de estos materiales, este trabajo está enfocado a buscar una metodología que nos permita la clasificación y búsqueda de este tipo de documentos utilizando minería de textos.

El primer paso consiste en la utilización de una metodología que nos ayude a buscar un proceso de operación para poder extraer la información y tener una posible solución al problema que planteamos relacionado a la extracción, clasificación y búsqueda de información. La metodología Crisp-DM se encuentra definida en un modelo jerárquico de procesos, esta metodología define un ciclo de vida de los proyectos de extracción de información. Las fases son: Análisis del problema, análisis de los datos, preparación de los datos, modelado, evaluación y explotación. Claramente estas fases difieren de las fases definidas para un proyecto de desarrollo de software clásico, esto es debido a la estructura que normalmente vemos en un texto, no tiene patrones definidos como lo es en otros problemas clásicos[7].

El análisis del problema nos ayuda a entender nuestro objetivo principal y lo que necesitamos buscar en los artículos que vamos a clasificar, esto nos va a ayudar a obtener una definición clara de nuestro problema y así buscar un diseño que nos permita lograr los objetivos

planteados en un inicio. El análisis de los datos comienza con la recolección inicial de datos y procede con las acciones necesarias de aprendizaje semiautomático para identificar aquella información que nos será útil en nuestro proceso así como detectar nuestros primeros subconjuntos de información y elaborar reglas preliminares de análisis.

La preparación de los datos cubre todas las actividades para poder obtener nuestra primer arquitectura propuesta con nuestros datos iniciales, procesos a implementar, y algunas veces podemos tener esta base sin un orden preestablecido esto puede incluir procesos relacionados a la limpieza de datos, preparación de información así como algoritmos de búsqueda y clasificación necesarios para la implementación.

El modelado ayuda a la selección de técnicas específicas para nuestra clasificación y búsqueda, incluyendo desde filters, wrappers o bien clasificadores específicos, este paso puede contener distintas técnicas para el mismo problema y cada una de las técnicas a implementar necesita diferentes requisitos, en este proceso es cuando se puede establecer si nuestro trabajo necesita adaptaciones o debe volver a trabajarse desde el primer punto relacionado al análisis del problema. Los métodos utilizados para el análisis de la información obtenida por el proceso de clasificación y búsqueda consisten en la parte final de los modelos encontrados en la literatura. Existen métodos de evaluación utilizados como es el método “leave-one-out”[8] el cual consiste en la revisión del contenido utilizado para la generación del motor de búsqueda y clasificación. Este método ayuda a revisar de manera detallada mediante el descarte de uno de los elementos y tratando de auto clasificarlo para así obtener el resultado, una vez hecho esto para todo el contenido de la base de conocimiento se determina la eficiencia del clasificador esto puede ayudar a obtener una conclusión que diga cuál es el proceso de modelado de mayor calidad. La fase de explotación e información está enfocada a una implementación final utilizada en un ambiente real de negocio[9].

La minería de textos es una tarea que busca clasificar un segmento de texto a un área predefinida de conocimiento, este proceso puede realizarse de diferentes maneras. La distribución de las áreas de conocimiento es un área importante en la clasificación del conocimiento ya que estas determinaran el comportamiento que tendrá nuestro algoritmo de clasificación. La gran mayoría de los algoritmos trabajan basándose en el peso que tienen en comparación con otro documento utilizando como base aprendizaje semiautomático, la selección de la categoría a la cual pertenece se basa según los métodos asignados por el

algoritmo acorde a los valores que se utilicen en el método de clasificación es lo que determinara la calidad de la clasificación.

La base que se utiliza en algunos algoritmos de clasificación consiste en tomar palabras de diferente tamaño y asignarle valores que representen su contenido, este algoritmo es una variante a otro aplicado el cual busca utilizar palabras aisladas que representan segmentos de una frase pero que son clave para la clasificación. Un algoritmo utilizado actualmente es el de extraer con frases del documento y realizar un cálculo estadístico que represente el valor que representa una frase dentro de un documento para clasificar. Existe un algoritmo que asume la independencia de las frases contenidas en un documento, esto nos ayuda a establecer valores independientes que no tienen relación entre ellos pero que debido a la clasificación asignada de una manera previa mediante aprendizaje semiautomático nos ayuda a establecer la clasificación que se debe aplicar[10].

El presente trabajo de tesis busca implementar un clasificador bayesiano utilizando Crisp-DM como metodología y minería de datos apoyándose en los valores obtenidos mediante Bayes para así poder establecer nuestra base de conocimiento en las áreas de programación, base de datos y sistemas operativos. Esta arquitectura no solo nos permitirá la clasificación también podremos realizar búsquedas utilizando el motor de clasificación bayesiano.

Capítulo 1 Método de Investigación

1.1. DESCRIPCIÓN DEL CONTEXTO DE LA SITUACIÓN PROBLEMÁTICA DE INVESTIGACIÓN.

La utilización de motores de búsqueda para la localización de información[1] ha crecido de manera constante en base a las necesidades de los usuarios generando un efecto de bola de nieve, donde toda la información está disponible en diferentes sitios web, incluyendo información que no es útil o relevante, pero también está incluida información de interés científico hay que recordar que no toda la información se encuentra por default en revistas de investigación especializada como las mencionadas a continuación Revista Especializada en ciencia (SPRINGER), Revista de Investigación Especializada en la investigación de avances en tecnología(IEEE) o Artículos reconocidos de investigación y divulgación (Journal).

La dispersión de la información es ocasionada cuando se tienen múltiples repositorios que contienen información relevante lo cual genera necesidades al momento de estar buscando recursos para agregar referencias, este tipo de búsquedas generalmente son de carácter específico donde se tiene un problema que necesita resolverse utilizando información relevante con el fin de obtener bases científicas para innovar en algún trabajo de investigación[11].

Actualmente existen sitios de búsqueda funcionando como metabuscadores[12] que ayudan a revisar la información que lanzan los otros buscadores[1], pero realizan búsquedas de manera normal , solo se encargan de presentar resultados en pantalla funcionando como interfaz de búsqueda en múltiples buscadores web.

Los buscadores académicos han evolucionado según las necesidades del sector de investigación. Ejemplos actuales de estos buscadores son Google Académico, seguido por Scirus, y Scienceresearch Microsoft AcademicSearch, entre otros. En general su funcionamiento depende de los resultados que éstos obtengan de los buscadores comerciales[13], para posteriormente recurrir a su propia generación de índices donde almacenan información sobre artículos y principalmente sobre la información que contienen las bases de datos de buscadores comerciales, donde si existe la posibilidad de visualizar el artículo de investigación y en caso de no estar disponible envían al usuario al sitio donde se puede comprar el artículo. Este tipo de situaciones han sido revisadas posteriormente planteando un buscador inteligente

el cual en base al uso de ontologías y clasificación de la información que realizara una búsqueda de información a través de diferentes motores de búsqueda a través de la web[11].

El uso de agentes inteligentes[12] se limita a la extracción de información de múltiples sitios web así como la generación de un propio índice de búsqueda que ayude a un mejor funcionamiento de un buscador web , con varios sitios disponibles con dicha tecnología como WebSeeker(www.bluesquirrel.com), Copernic(www.copernic.com), EZSearch(www.americansys.com) o LexiBot(www.lexibot.com).

El presente trabajo propone la utilización de un agente inteligente que ayude a la identificación aproximada en el texto a buscar utilizando patrones predefinidos, así como la implementación de un algoritmo de clusters[3] para las consultas realizadas dentro del manejador de base de datos mysql (manejador de base de datos libre que permite el uso de multihilo, multiconsulta y multiusuario) con el fin de obtener de artículos científicos de investigación.

1.2 RELEVANCIA Y JUSTIFICACIÓN DE LA INVESTIGACIÓN.

La implementación de una herramienta que permita una mejor búsqueda de artículos de investigación, actualmente solo se ha presentado trabajo hasta nivel de agentes inteligentes con interpretación de lenguaje natural básico. La propuesta de este trabajo es el establecimiento de una arquitectura conceptual que permita una mejor optimización de las búsquedas dentro de un entorno web, con esto se podrá tener una arquitectura que sea fácil de implementar con fines de investigación, así como un desempeño optimo al momento de realizar búsquedas.

El principal motivo de la mejora en la búsqueda de artículos es ayudar a tener un mejor proceso de localización de la información, este proceso no solo es implementar una búsqueda de artículos de investigación, consta también de aplicar técnicas de minería de datos que ayuden a la correcta clasificación de la información para así este mismo motor utilizarlo en la localización utilizando el texto de búsqueda que el usuario introduce. Las principales ventajas son las siguientes:

- Generar una base de conocimiento que ayude a la correcta clasificación de artículos de investigación en el área de base de datos, sistemas operativos y programación.

- Implementar un algoritmo K-Means perteneciente al área de minería de datos para la Clusterización de las categorías propuestas en el presente trabajo para una mejor estructura de la información.
- Implementar un Filter y Wrapper pertenecientes al área de minería de textos, son las herramientas utilizadas para la clasificación de texto así como comparar el rendimiento de ambas , utilizando al menos dos Meta heurísticas en el funcionamiento del wrapper.
- Implementar una herramienta que permita la clasificación del documento así como establecer los procesos de filter y wrapper.
- Generar una serie de resultados estadísticos que demuestren cual técnica tiene mejor desempeño.

La implementación de este tipo de técnicas, comparadas para la clasificación y búsqueda de información tienen diferentes implantaciones para auxiliar a diferentes investigadores no solo a realizar búsquedas, sino también a alimentar repositorios con distintos tipos de contenidos siempre y cuando la base de conocimiento tenga la información correcta, esto le permitirá la explotación de la herramienta.

La implementación de una arquitectura que permita la generación de búsquedas a través de un esquema organizado utilizando la metodología CRISP-DM donde en base al análisis de la información se genera una serie de atributos significativos para el funcionamiento del clustering que planea implementarse.

1.3 DESCRIPCIÓN GENERAL DEL ENFOQUE DE INVESTIGACIÓN.

El tipo de investigación a utilizar será cuantitativo realizando una comparación entre dos métodos para la localización y clasificación de artículos de investigación mediante la utilización de un filtro o mediante un wrapper. Las variables a considerar para el análisis de la investigación son las siguientes:

- Tiempos de respuesta para la clasificación del artículo de investigación.
- Tiempos de respuesta para la localización del artículo de investigación.
- Mediante aprendizaje semiautomático revisar la calidad de los resultados obtenidos.
- Concluir mediante los tiempos obtenidos que es mejor si un filtro o un wrapper.

1.4 PROBLEMA DE INVESTIGACIÓN.

La localización de artículos de investigación en un entorno web donde existen diversos buscadores orientados a una búsqueda genérica de información[13] genera la necesidad de diseñar una arquitectura que permita la localización de artículos de investigación donde se propone el uso de un algoritmo que permita la identificación de patrones en el texto de búsqueda así como un algoritmo para generar clusters con minería de datos.

Este planteamiento se considera válido[14] debido a la combinación de técnicas que planean utilizarse para la solución del problema en las cuales incluye una base de conocimientos que permitirá la relación entre el texto introducido y el área en la cual se relaciona para poder determinar el área de interés al cual está dirigido, así como un algoritmo que permita generar clusters como minería de datos para poder mejorar el tiempo de respuesta en las búsquedas de artículos de investigación.

La actividad de búsqueda de literatura científica[15] es una actividad crítica en el proceso de cualquier investigación[13], pero que suele consumir grandes cantidades de tiempo, el uso de buscadores para localizar artículos de investigación[11] en el área de lenguajes de programación que permita la identificación básica de patrones en el texto introducido así como la implementación de un algoritmo de minería de datos para ayudar a disminuir el tiempo de respuesta en la búsqueda dentro de la base de datos(mysql) para la localización de artículos científicos, así como una implementación a nivel prototipo que permita acceder desde un navegador(firefox), esto con el fin de facilitar una posible expansión hacia otros dispositivos vía Web.

Es necesario el desarrollo de un prototipo que permita el análisis del funcionamiento para realizar pruebas en base a la cantidad de resultados precisos relacionados al tipo de búsqueda realizada así como a la relación obtenida en artículos. A partir de la base de conocimientos obtenida, deberá ser tomada como partida, para determinar patrones dentro de una palabra capturada y así deducir el peso que se le dará al momento de enviar dicha información al algoritmo de minería de datos.

Desarrollar una arquitectura que permita la implementación del manejo y uso de base de conocimientos adecuando una correcta interpretación de patrones relacionados a cada frase,

aplicando un filtro correcto de clasificación. Dicha arquitectura aplicada, formara una búsqueda en forma de consulta dentro de una base de datos(mysql), obteniendo contenidos relacionados a los artículos científicos que se especificaron en la consulta con el algoritmo de minería de datos para obtener un mejor rendimiento en el tiempo de consulta al obtener la información dentro del acervo de contenidos digitales de investigación[16].

1.5 PREGUNTA Y OBJETIVO DE INVESTIGACIÓN.

Objetivo General:

- Generar una aplicación utilizando CRISP-DM que permita la búsqueda de artículos de investigación de manera óptima en tiempo y resultados.
- Utilizar la metodología CRISP-DM para la clasificación de artículos de investigación de manera óptima en tiempo y resultados.

Objetivos Específicos:

- Generar una base de conocimiento en el área de programación, base de datos y sistemas operativos. Estableciendo un peso específico para las palabras almacenadas en la base de conocimiento.
- Crear un algoritmo que pueda recorrer el documento en formato pdf para analizar el contenido comparándolo contra la base de conocimientos para establecer la categoría a la cual pertenece.
- Implementar el algoritmo k-means para la clusterización de los documentos almacenados según la categoría establecida respetando el proceso CRISP-DM.
- Implementar un clasificador bayesiano para determinar la categoría de la búsqueda o clasificación.
- Establecer información estadística en base al rendimiento y calidad de los resultados.

¿Es posible representar por medio de una arquitectura conceptual una búsqueda que permita el análisis de patrones en el texto de búsqueda para implementar clusters y localizar efectivamente artículos de investigación?

¿Se puede mejorar el tiempo de respuesta de un proceso de búsqueda mediante el uso de clusters y la identificación de patrones en el texto introducido?

¿Es posible establecer una arquitectura que permita una rápida búsqueda de artículos de investigación implementando clusters en las búsquedas?

¿Se puede implementar un correcto esquema de patrones a identificar en el texto para el entendimiento del lenguaje natural para interpretarlo en forma de consulta?

¿Es posible implementar un clasificador bayesiano para optimizar el rendimiento en las búsquedas dentro del acervo de artículos de investigación?

1.6 HIPÓTESIS O PROPOSICIONES DE INVESTIGACIÓN.

H. 0. No es factible el diseño e implantación de un Filter que permita la localización y clasificación de artículos de investigación con una calidad en los resultados analizada en base a aprendizaje semiautomático utilizando como base la metodología CRISP-DM.

H. A. Es factible el diseño e implantación de un Filter que permita la localización y clasificación de artículos de investigación con una calidad en los resultados analizada en base a aprendizaje semiautomático utilizando como base la metodología CRISP-DM.

Capítulo 2 Marco Teórico

2.1. Descubrimiento del conocimiento.

El terminó Descubrimiento del Conocimiento (KDD, iniciales de *Knowledge Discovery in Databases)* fue inicialmente usado en 1989 se refiere a todo proceso de extracción de conocimiento a partir de una base de datos y marca un cambio en el paradigma en el que lo importante es el conocimiento útil que seamos capaces de descubrir a partir de los datos. El concepto KDD continua desarrollándose desde la intersección de la investigación de áreas tales como bases de datos, aprendizaje automático, reconocimiento de patrones, estadística, inteligencia artificial, razonamiento visualización. Estos sistemas utilizan teorías, algoritmos y métodos de todos estos campos. El proceso de KDD es interactivo e iterativo conteniendo los siguientes pasos[17]:

- Comprender el dominio de aplicación: este paso incluye el conocimiento previo y las metas de la aplicación.
- Extraer la base de datos objetivo: obtener los datos, evaluar la calidad de los datos y utilizar un análisis exploratorio de los datos para familiarizarse con ellos.
- Preparar los datos: incluye limpieza, transformación, integración y reducción de datos. Se intenta mejorar la calidad de los datos a la vez que disminuir el tiempo requerido por el algoritmo de aprendizaje.
- Minería de datos: como se ha diseñado anteriormente, esta fase es fundamental en el proceso constituye por una o más de las siguientes funciones clasificación, regresión, clustering, resumen, recuperación de imágenes, extracción de reglas, etc.
- Interpretación: explicar los patrones descubiertos, así como la posibilidad de visualizarlos.
- Utilizar el conocimiento descubierto: hacer un uso del modelo creado.

La minería de datos se encarga principalmente del descubrimiento de conocimiento para encontrar información no trivial que es potencialmente útil debido a que se encuentra en grandes repositorios de información su localización resulta complicada. La minería de datos es un área multidisciplinar donde convergen diferentes paradigmas de computación como son la construcción de árboles de decisión, la inducción de reglas o patrones, redes neuronales artificiales, el aprendizaje basado en instancias, aprendizaje bayesiano, programación lógica,

algoritmos estadísticos, etc. Las principales tareas y métodos de la minería de datos son : clasificación, agrupamiento, estimación, modelado de dependencias, visualización y descubrimiento de reglas[18]. La minería de datos es en realidad uno de los pasos que comprenden el proceso de descubrimiento de conocimiento, que está compuesto por:

- Preprocesamiento: Consiste en la extracción de los datos, limpieza de datos, desratización, selección de los atributos e integración de datos.
- Minería de datos: Consiste en la selección de los algoritmos de minería de datos a utilizar y la aplicación de dichos algoritmos sobre los datos.
- Postprocesamiento: Consiste en la interpretación, evaluación de los resultados obtenidos y la utilización del conocimiento descubierto.

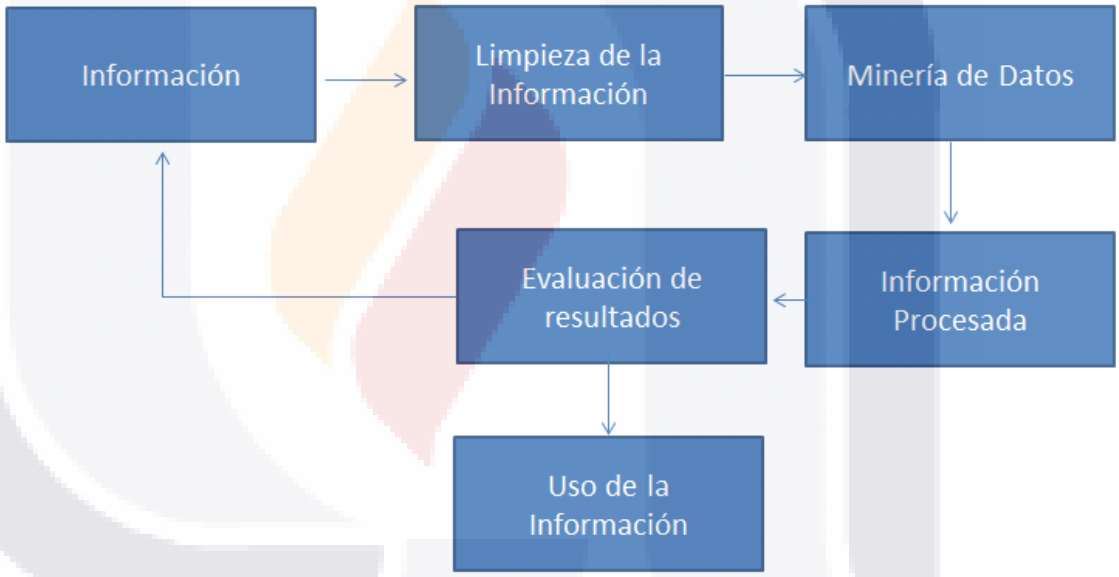

Figura 1. Modelo de Minería de Datos[19].

El área de minería de textos pertenece a una área similar a la minería de datos, solo que se encarga del descubrimiento de información en grandes contenidos de texto y automáticamente se encarga de localizar e identificar patrones en el texto así como relaciones. Es un área relativamente nueva la cual gracias a las grandes cantidades de información que se pueden localizar en Web y documentos digitales. La minería de textos es un área de investigación que utiliza varias técnicas relacionadas a la minería de datos como el procesamiento de lenguaje natural, aprendizaje automático y extracción de información. La

gran diferencia entre minería de datos y minería de textos consiste en el tipo de contenido que se analiza debido a que al trabajar lenguaje natural resulta complicado el trabajar con información que no tiene una estructura previa a diferencia de las tablas previamente organizadas en minería de datos[20].

2.2. Clusterizacion.

El término clusterización se refiere al proceso de particionar[4] o agrupar una serie de patrones o información dentro de clusters con información similar, este proceso implica la separación de los datos en un determinado número de grupos con patrones similares, y aquellos que sean diferentes se agrupan según sus coincidencias en su información. La utilización de clustering ha sido aplicada a una gran variedad de dominios incluyendo redes neuronales, inteligencia artificial y estadística.

Existe una variedad de algoritmos propuestos en la literatura para clustering el algoritmo K-Mean[21]s es el que tiene una mejor efectividad en la generación de clustering ofreciendo mejores resultados para problemas aplicados. La implementación de K-Means implica la selección de cierto número de patrones para la correcta identificación de los clusters.

El uso de clusters o clustering implica la clasificación de diferentes grupos esto con el fin de generar grupos particionados que compartan alguna característica de esta manera se determina una medida de cercanía entre ellos creando grupos. El proceso de clustering compone los siguientes pasos[22]:

- Representando la información mediante vectores los cuales son calculados utilizando la información que contiene cada dato a incluir en el cluster, esto puede incluir de manera opcional la extracción de información disponible o bien selección.
- Definir la unidad de medida que ayudará a la clasificación de la información.
- Existen diferentes métodos para la aproximación en la agrupación de clusters, pueden definirse de la siguiente manera[23]:
    - Algoritmos Jerárquicos: Permite la creación de una jerarquía o bien la destrucción de la misma, esto puede dividirse en algoritmos aglomerativos que inician con una mezcla de clusters con ciertos valores de manera aleatoria hasta que logran llegar a un acercamiento deseado en los clusters. Mientras

que el algoritmo divisivo genera en un inicio un solo cluster que se empieza a dividir hasta obtener el resultado óptimo.

- Algoritmos Particionales : Este tipo de algoritmo genera en un solo proceso como deben terminar los clusters

- Abstracción de los datos.
- Evaluación de los datos.

Este tipo de comportamiento utilizando clustering puede clasificarse como aprendizaje semiautomático, donde depende del tipo de algoritmo de clasificación que se implemente y el tipo de medidas utilizadas para alimentarlo , debido a que depende del tipo de información que se asigne ya que si se agregan un número indefinido de variables puede disminuir el rendimiento del algoritmo implementado. Esto es debido a que no todos los atributos son relevantes en la clasificación de información, se deben seleccionar aquellos que representen valores relevantes para lograr una correcta clasificación.

El uso de algoritmos jerárquicos mejor conocidos como Single Link(SL), Average Link(AL) y Complete Link (CL), se describen de la siguiente manera:

- Single Link : En cada paso se unen los dos grupos cuyos elementos más cercanos tienen la mínima distancia.
- Average Link: En cada paso se unen los dos grupos tal que tienen la mínima distancia promedio entre sus puntos.
- Complete Link: En cada paso se unen los dos grupos tal que su unión tiene el diámetro mínimo o los dos grupos con la menor distancia máxima entre sus elementos.

Algoritmo Chameleon

Es el algoritmo representativo del método jerárquico se compone de dos fases principales durante la primera fase construye el grafo de los k vecinos más cercanos y usa un algoritmo de particionamiento de grafo para agrupar los puntos en subgrupos[23].

Durante la segunda fase usa un algoritmo jerárquico aglomerativo para encontrar los clusters genuinos combinando repetidamente estos subgrupos. En esta segunda fase determina el par de subgrupos más similares tomando en cuenta su interconectividad y cercanía, estas expresan

las características internas de los subgrupos, el modelo no es estático si no que es capaz de adaptarse a las características internas de los subgrupos según estos van cambiando.

Existe la posibilidad de generar las relaciones entre atributos utilizando métodos de particionales en un número preespecificado de conglomerados k, y luego iterativamente se van reasignando las observaciones a los conglomerados hasta que algún criterio de parada (función de optimizar) se satisface (suma de cuadrados dentro de los conglomerados sea la más pequeña). Ejemplo de este método es el algoritmo K-Means, PAM, CLARA, SOM, todos conglomerados basados en modelos de mezclas gausianas.

Algoritmo K-Means(Mac Queen 1967)

Es un algoritmo creado para dividir de una forma fácil y simple una base de datos en k grupos que son definidos previamente[24].

La idea principal es definir los centroides uno para cada grupo para posteriormente tomar cada punto de la base de datos y situarlo en la clase de su centroide más cercano. El próximo paso es recalcular el centroide de cada grupo y volver a distribuir todos los objetos según el centroide más cercano. El proceso se repite hasta que ya no hay cambios en los grupos de un paso siguiente[25].

El problema con este algoritmo es que el método falla cuando los puntos de un grupo están muy cerca del centroide del otro grupo así como cuando los grupos tienen diferentes formas y tamaños.

Algoritmo CURE

Constituye un algoritmo híbrido entre los enfoques jerárquico y particional que trata de emplear las ventajas de ambos y eliminar las limitaciones. En este algoritmo en lugar de usar un solo punto como representante de un grupo se emplea un número c de puntos representativos del grupo. La similaridad entre dos grupos se mide por la similaridad del par de puntos representativos más cercanos, uno de cada grupo[23].

Para tomar los puntos representativos se seleccionan los c puntos más dispersos del grupo y los atrae hacia el centro del mismo por un factor de contradicción alpha , en cada paso se unen los

dos grupos más cercanos y una vez unidos se vuelve a calcular el centro del grupo y los c puntos representativos[24].

Con este algoritmo se encuentran grupos de diferentes tamaños y formas con este método de sacar c puntos representativos y atraerlos hacia el centro del grupo CURE maneja los puntos ruido y outliers presentes en la base de datos.

El principal objetivo es estandarizar la información que se utilizará identificando atributos que sean clave para la implementación del algoritmo y así pueda disminuir el número de atributos que tengan eliminando información redundante o poco relevante, de esta manera se tendrán los atributos relevantes para la correcta aplicación del algoritmo.

Una vez definidos los atributos se debe iniciar con los pasos a seguir para realizar la búsqueda para esto será necesario establecer un punto para el inicio de la búsqueda, una estrategia para recorrer los registros, una función para evaluar los atributos clave y un criterio para detener la búsqueda en caso de haber encontrado el correcto[21].

Punto de Inicio: En primer lugar es necesario establecer el inicio de nuestra búsqueda, se puede iniciar con el conjunto completo de todos los atributos y conforme avanzamos en el proceso de la metodología CRISP-DM ir eliminando atributos. Existe la otra opción donde se inicia sin ningún atributo y conforme se va avanzando se van añadiendo los atributos que se necesiten.

Estrategia de Búsqueda: Se debe establecer el método que se aplicará para la búsqueda de los elementos una posibilidad es la de buscar por todos los elementos, si bien este método nos asegura la localización del elemento que necesitamos es poco recomendable si el número de elementos que tenemos es elevado.

Para un conjunto de n atributos el espacio de búsqueda es de 2^n -1; para la selección del subconjunto de m atributos de forma exhaustiva es necesario comprobar los subconjuntos.

Para evitar el recorrido de todo el espacio se han definido estrategias que ayudan a determinar un subconjunto de atributos que no aseguran el óptimo pero que tienen un valor aproximado con respecto a la función de evaluación utilizada. Las más utilizadas son las secuenciales y las Aleatorias[26].

La estrategia aleatoria se basa en visitar diferentes regiones del espacio sin tener un orden predefinido, de esta manera se espera obtener el subconjunto óptimo de atributos.

La estrategia secuencial también conocida como heurísticas funciona mientras se va ejecutando añadiendo nuevos atributos a los ya seleccionados. También pueden llegar a eliminarse atributos que hayan sido seleccionados en el inicio del proceso. Existen dos métodos dentro de la estrategia secuencial, el primero es selección secuencial hacia adelante (Forward Selection (FS)) y el segundo es eliminación secuencial hacia atrás (BackwardElimination (BE)).

El algoritmo para la búsqueda Forward Selection es el siguiente:

1. Calcular los valores para cada atributo independiente
2. Seleccionar el atributo que mejor valor parcial obtiene
3. Calcular todos los valores para todas las combinaciones con los atributos restantes
4. Volver al paso 2

El algoritmo para la búsqueda en BackwardElimination:

1. Calcular los valores parciales para cada combinación de n-1 atributos.
2. Eliminar el atributo que menor valor parcial obtiene.
3. Calcular todos los valores para todas las combinaciones de n-1 atributos con los restantes
4. Volver al paso 2

Función de Evaluación: Es necesario establecer una función de evaluación para cada subconjunto de atributos, esta medida de evaluación estará definida para un conjunto de atributos y deberá medir la capacidad discriminante del conjunto de atributos y deberá medir la capacidad para distinguir entre las diferentes clases de atributos definidas en el problema.

Existen dos principales algoritmos orientados a las medidas de evaluación conocidos como filtro y envolvente. En general los algoritmos de filtro el proceso de selección se realiza como un pre proceso independiente al proceso de clasificación, esto en función a las características generales del conjunto de entrenamiento se seleccionan o filtran características y se excluyen otras.

Los algoritmos envolventes hacen uso del proceso de clasificación para evaluar la calidad de cada conjunto de atributos seleccionados en cada momento. En este caso el algoritmo de aprendizaje se ejecuta sobre los datos de entrenamiento y se utilizan parámetros de evaluación para analizar el conjunto de atributos.

Criterio de Parada: Se debe establecer un criterio de parada que permita determinar cuándo se ha encontrado el conjunto de atributos para los que la función de evaluación da el valor óptimo. El no establecer un criterio de parada supone que se recorrerá todo el espacio de la búsqueda con lo que implica el tiempo de espera computacional, Esto en Ciencias de la Computación se le conoce aunque de manera parcial, en métodos, como "finitud de programa o finitud del algoritmo"[27].

2.3 Clasificadores

El interés en el la búsqueda y clasificación de artículos de investigación se ha incrementado debido al comportamiento que se ha visto reflejado en las bibliotecas digitales e internet, en este contexto se han analizado consultas realizadas a bibliotecas digitales y el registro de esas consultas fue almacenado para posteriormente construir un registro de comunidades de usuarios con intereses similares utilizando minería de datos con el fin de que estas comunidades puedan mejorar su acceso a la información.

La clasificación trata de encontrar las características que identifican a un grupo para ser clasificado dentro de cierta clase. Este conocimiento puede ser utilizado para entender el comportamiento del sistema que género los datos y de esta forma predecir la clase a la que pertenecerá una nueva instancia, entre los algoritmos de clasificación se encuentran:

Análisis discriminante. El método utilizado para este algoritmo es determinando la localización optima de una línea que actúa como límite entre los diferentes casos, este algoritmo busca la línea de tal manera que el margen de separación entre casos de diferente tipo sea máximo. Este tipo de algoritmo es muy fácil de implementar sin embargo no siempre se puede hacer este tipo de discriminaciones.

k-vecinos más cercanos. Si se tiene conocimiento de elementos contenidos en un grupo con características similares, este algoritmo forma un grupo de k individuos de acuerdo a sus características. Cuando aparece un nuevo individuo este es clasificado de acuerdo a la semejanza que presente con respecto a los grupos clasificados previamente.

Redes neuronales. Este tipo de algoritmo intenta emular el funcionamiento de los cerebros de los seres vivos mediante capas de "neuronas" que utilizan funciones matemáticas con un comportamiento previamente establecido. Existe una capa de entrega seguida de una o varias capas intermedias para finalizar en una capa de salida.

Arboles de decisión. Estos algoritmos usan reglas a partir de datos tratando de obtener una descripción más sintética que represente de forma más cercana los datos originales. Cuando se presenta un nuevo caso, se siguen las reglas extraídas por el algoritmo y se determina el grupo al cual pertenecen.

Vectores soporte. Estos algoritmos están relacionados con el análisis discriminante, utilizan técnicas de vectores de soporte para el pronóstico de acontecimientos.

Naive Bayes. Este método está basado en la teoría de la probabilidad, usa frecuencias para calcular probabilidades condicionales para calcular predicciones sobre nuevos casos. Esta técnica es tanto predictiva como descriptiva a pesar de ser simple ha sido desarrollada con éxito, produciendo buenos resultados en sus aplicaciones.

El algoritmo utilizado para nuestro clasificador y buscador será Naive Bayes , por sus buenos resultados y facilidad de adaptarse a diferentes tipos de problemas, uno de los principales motivos para seleccionar este algoritmo es debido a que utiliza como apoyo probabilidades condicionales para calcular los posibles resultados. Este tipo de funcionamiento se adapta a un entorno donde se va a generar un clasificador que será el encargado de clasificar lenguaje natural el cual es complejo clasificar y localizar.

2.4 Minería de Textos

La disponibilidad de una gran cantidad de información en forma de texto se convierte en una forma habitual de encontrar información en distintos lugares como es en la web, librerías digitales, documentación técnica, información médica, etc. Esta información textual constituye recursos que pueden ser explotados[12] obteniendo el conocimiento extraído de manera directa del texto que viene dentro de cada documento para ser manipulado este tipo de manipulación es llamada minería de textos es complicado realizar este tipo de extracción debido a la riqueza y ambigüedad del lenguaje natural debido a que existe una gran cantidad de información que van desde información contenida en bases de datos hasta información

disponible vía Web. En este contexto un análisis manual y efectivo de información no es la mejor opción debido a la cantidad de tiempo que llevaría revisar toda la información de manera manual.

La minería de datos tiene como principal objetivo obtener información a partir de los patrones[4] y tendencias que pueden observarse en grandes volúmenes de información estructurada, es decir información disponible en bases de datos relacionales frente a esto la minería de texto busca un mismo objetivo en archivos de texto no estructurados.

Los principales fabricantes de aplicaciones informáticas para la minería de datos promueven la imagen de la minería textual como una disciplina complementaria a la primera y han acoplado a sus programas diferentes módulos para la extracción y análisis de textos.

La agrupación de documentos similares o clustering[28] consiste en unir documentos entre los que existe cierta similitud, la cual se establecerá a partir de la terminología utilizada por los autores en la redacción de los textos. Esta funcionalidad de las aplicaciones de minería de textos aplica una de las técnicas características de la recuperación textual mediante el clustering.

La clasificación automática[2] es un proceso de clasificación que pretende asignar un documento a una clase o un tema definido con anterioridad, este proceso parte de un aprendizaje semi automático previo del programa que se encargará de la clasificación, de esta manera cuando el programa recibe un nuevo documento en base a la información que contiene podrá decidir a qué clase pertenece.

El proceso que se implemente para la clasificación del documento parte de un análisis inicial del documento donde permite extraer los principales temas o ideas tratadas en los documentos, no se trata de un proceso de clasificación automática, ya que no se pretende asignar un documento a una clase sino extraer un conjunto de términos que son representativos del contenido de los documentos, este proceso puede ser aplicando clustering relacionada a los términos encontrados buscando establecer información similar entre los elementos seleccionados mediante un proceso estándar que busca :

- Eliminar palabras que poseen menos de 3 caracteres.
- Eliminar palabras genéricas.

- Eliminar adverbios y adjetivos.
- Eliminar verbos innecesarios.

Este proceso nos ayudara a tener un documento pequeño con la información necesaria para la clasificación de esta manera podemos continuar con la Clusterizacion del documento donde se busca el acomodo de las palabras según el grupo al cual pertenezca[29].

Los pasos para lograr implementar una minería de textos[30] son los siguientes:

- Pre-procesamiento de los documentos, contendrá los términos principales para la clasificación, palabras vacías y ruido que pueda tener el documento.
- Identificación de nombres propios: Análisis sintáctico y gramatical de los textos.
- Representación de los documentos mediante la fórmula que ayudara a interpretar la similitud entre las frases de los documentos.
- Clustering, agrupación de los documentos similares.

La clasificación automática (wrapper) consiste en la utilización de una meta heurística que establecerá una serie de categorías de manera automática utilizando el documento como base para auto clasificar el tipo de documento que se encuentra recorriendo.

La clasificación mediante términos y conceptos (Filter) es una de las técnicas en las cuales se tiene una extracción de palabras clave y mediante el peso que se tiene por cada frase se establece una relación entre ambas frases para así determinar en que cluster se va a localizar[30].

El funcionamiento de la minería web puede ser dividido en tres diferentes tipos utilización de minería web, contenido en minería web y estructura de minería web. La implementación de minería web en contenidos es el proceso de descubrir información útil para obtener el contenido de una página web. El contenido de un sitio web normalmente consiste de información en forma de texto[5], audio o video , suele llamarse a este contenido como “minería de texto en web” por qué el contenido es en su mayoría información en texto, donde toda aplicación de técnicas de minería de texto web se basan en la extracción de información que se considera útil para obtener patrones así como descartar información que no es útil para

la clasificación este proceso es utilizando técnicas de análisis en el lenguaje natural que ayudan a establecer la obtención de información[31].

La minería de textos[4] web es un área que ha ido creciendo en base a las grandes cantidades de información que se encuentran almacenadas en formato de página web, correos electrónicos y reportes específicos. Los avances en minería web son gracias a la exploración e implementación de herramientas que ayuden al reconocimiento de patrones[1] en el texto donde la especificación de los patrones se basa en el establecimiento de estadísticas según los patrones encontrados. La aplicación de estos patrones genera cadenas de texto que ayudan al establecimiento de textos que definan procesos o tareas específicas.

El proceso de minería de texto precede primeramente de un proceso que ayuda a establecer una estandarización del texto tomando de base un segmento de cadena no estandarizado y pasándolo a información estructurada que facilite el tratado de la información. La extracción de la información utiliza un esquema xml[4] , archivo o base de datos que almacena elementos clave para la clasificación. La aplicación de filtros en la minería de textos depende del tipo de conocimiento que espera obtenerse si se aplica un filtro con atributos específicos y cerrados puede obtenerse un conocimiento filtrado.

El PhD Mutlu Mete[32], presenta un concepto relacionado a las secuencias que existen entre palabras que ocurren de manera constante unidas en un texto a pesar de que dichos conceptos son diferentes en otros lenguajes desde el punto de vista de minería de textos se pueden establecer reglas de asociación donde se lograría la generación de un framework para poder implementarlo.

La utilización de minería de textos inicia con una limpieza de los contenidos con el fin de eliminar aquellas frases que no sean necesarias o primordiales para nuestro análisis, una vez que tenemos lista nuestra información continuamos con la extracción de conceptos que sean más significativos para obtener patrones determinando que la frecuencia con la cual ocurre en el documento[32]. Una vez obtenemos nuestro primer listado de reglas a aplicar al contenido se busca en el contenido aquellas palabras significativas, se extraen analizan y se determina hacia qué tipo de contenido está orientado.

La minería de textos es la ciencia que trata del estudio y análisis del lenguaje natural según Gelbukh y Bolshakov en 1999, esta ciencia es una combinación de dos ciencias más grandes; la

lingüística que estudia las leyes del lenguaje humano y la inteligencia artificial que investiga los métodos computacionales para el manejo de sistemas complejos. El principal problema de esto es la comprensión del lenguaje natural, es decir la transformación del lenguaje en una red semántica que nos ayude a construir nuestro procesador y así interpretar el significado de las frases[33].

La investigación del procesamiento de textos define el descubrimiento de patrones interesantes y nuevos conocimientos en una colección de textos, es decir la minería de texto es el proceso encargado del descubrimiento de conocimientos que no existían explícitamente en ningún texto de la colección pero que surgen de relacionar el concepto de varios de ellos acorde a Hearst y Kodratoff en 1999[33].

La minería de texto según Marti Hears de la UC Berkley determina que es el descubrimiento de información de una manera automática extrayendo información de diferentes ubicaciones, donde el elemento clave es la relación entre las frases que están siendo exploradas para la utilización en otras aplicaciones como podría ser una búsqueda web, donde existen diversos conceptos similares que podríamos obtener en una búsqueda. La minería de textos es una variación de un campo llamado Minería de Datos, la cual busca encontrar patrones interesantes en bases de datos, la principal diferencia entre minería de datos y minería de textos es que en una los patrones son extraídos de lenguaje natural mientras que en la otra desde información estructurada en bases de datos, este tipo de variante genera un problema importante ya que a la fecha no tenemos programas que nos ayuden a “leer” la información que está contenida en un texto. Este tipo de funciones son tratadas como comprensión del lenguaje natural donde se busca extraer pequeñas partes de conocimiento que nos ayuden a determinar qué es lo que estamos buscando y/o analizando[34].

La búsqueda automática de información es uno de los últimos retos en la extracción de información y análisis de texto, actualmente hay indexación de documentos pero se hace en momentos que no se está utilizando el documento y pertenece a un área de aprendizaje supervisado. Dentro de las categorías de lenguaje natural que se busca explotar es el de contenidos en voz donde existe potencialmente una gran cantidad de información que se podría obtener mediante técnicas que analizan la información de voz que se está recibiendo y automáticamente buscan información que sea relevante y ayude a clarificar el valor de la información que se recibe en ese momento[35].

La información que actualmente es almacenada en forma de texto normalmente se encuentra sin una estructura como por ejemplo artículos en sitios de investigadores, comentarios acerca de algún tema en específico entre otros son normalmente almacenados en forma de texto con libre acceso hacia ellos, este tipo de información disponible se ha vuelto en una área de oportunidad donde se puede buscar el análisis automático del posible conocimiento almacenado en esos documentos libres utilizando como lo definió Dozier en el 2003 minería de textos el cual es un nuevo campo de investigación que busca la obtención de relaciones o patrones entre el texto y su posible significado. Los objetivos que se presentan en la minería de textos es lo siguiente[36]:

- La extracción de la información es una técnica que acorde a lo contenido en el texto, toma fragmentos que son mapeados en un template el cual tiene como base la obtención de un área específica de contenido, la técnica fue propuesta por Cowie y Wilks en el 2000.

- La suministración del texto envuelve la identificación y organización de textos que son relacionados acorde a la información que es contenida en largos documentos.

- La clasificación del texto busca la organización de textos en una taxonomía que ayude a tener búsquedas de manera más eficiente, además ayuda a la asignación de códigos o clasificadores que ayudan a analizar textos completos de una manera más sencilla.

- El clustering en texto involucra la clasificación automática de los documentos en base a grupos que ayudara a determinar qué características tienen en común.

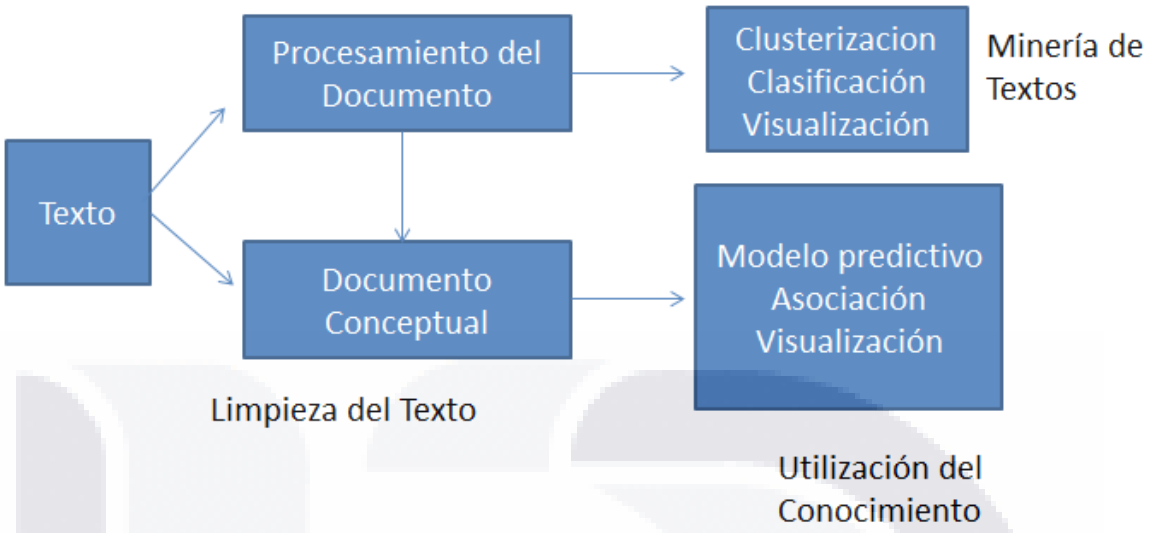

Figura 2. Framework utilizado para Minería de Textos[37].

La figura 2 muestra el framework que es utilizado comúnmente por la mayoría de los trabajos orientados a minería de texto donde se tiene un proceso intermedio que en primera instancia limpia el texto a analizar para así poder obtener el contenido principal a nivel conceptual del texto que se va a analizar, de esta manera se podrá obtener un documento puro que será utilizado para la utilización del conocimiento, donde puede abarcar tareas como clasificación, Clusterizacion y visualización en el proceso de limpieza del documento. En el área relacionada al documento conceptual podemos establecer un modelo predictivo, asociación entre los patrones encontraros y visualización del propio documento[37].

El primer paso para obtener una clasificación del texto consiste en la interpretación previa del texto, pero para este paso es necesario aplicarle un filtro que ayude a obtener un documento más estructurado para su correcto análisis, una vez obtenida la información estructurada es necesario aplicar el modelo de clasificación propuesto y evaluar sus resultados.

El primer método para clasificación de información es KNN (K NearestNeighbor) el cual consiste en aplicar una serie de métricas para generar una estructura la cual ayudara a la clasificación en base a atributos establecidos previamente. El método funciona mediante el cálculo de las distancias entre los vecinos más cercanos y lo relaciona según el valor arrojado por el proceso que determinara el clasificador que se aplicara. En el año de 1992 KNN fue aplicado para la clasificación de artículos en Massand, en 1999 se estableció un acercamiento a la clasificación

de artículos llegando a la conclusión de que era un método muy recomendable para segmentar información. En el 2002 Sebatiani evaluó KNN como un algoritmo sencillo y competitivo. La principal desventaja de aplicar KNN es que en caso de tener una gran variedad de posibilidades por clasificar se deberá tener una gran base de conocimientos que ayude a la correcta clasificación de la información[38].

El segundo método utilizado en diversas investigaciones es NaiveBayes (NB) este método implica la clasificación del texto mediante la aplicación de una regla de probabilidad condicional donde se evalúan todos los atributos contenidos en el texto y se analizan de manera individual obteniendo la importancia de cada elemento del texto. Tiene una ventaja en ejemplos estructurados de manera correcta ya que obtiene un mejor rendimiento sobre el algoritmo KNN.

El siguiente método de clasificación es mediante la implementación de árboles de decisión los cuales usan como base una máquina predictiva (machine-learning) el modelo decide en base a una serie de términos obtenidos previamente cual será el camino a tomar según el árbol predefinido previamente. El árbol de decisión establecido depende en gran parte de la información con la cual fue alimentado donde según el tamaño será el tipo de decisiones que tome para la clasificación de la información, entre más robusto sea nuestro árbol mejor clasificación obtendremos.

La SVM (Support Vector Machine) ha sido uno de los mejores clasificadores de textos utilizando como entrada una gran cantidad de documentos de texto, este tiene un funcionamiento óptimo hablando en términos de medidas para la clasificación de texto y existen otras medidas como el tiempo en la selección de atributos, aprendizaje para la selección de clasificadores y pruebas a realizar en el clasificador. Mediante este proceso se analiza el texto para la obtención de los clasificadores y patrones para seleccionar los atributos que se utilizaran para clasificar el texto sin realizar un filtrado previo mediante un aprendizaje semiautomático, propone un proceso completamente automatizado sin tomar en cuenta el tiempo de desempeño.

La selección de atributos es un paso para el pre procesamiento de la información existen dos métodos genéricos los cuales son: Filter y wrapper. En el Filter se trabaja mediante un establecimiento de medidas para cada atributo que es leído en el texto esto sin la utilización de

algún algoritmo para el autoaprendizaje en el texto. En el wrapper se utiliza un algoritmo de aprendizaje el cual según el texto que es analizado se obtiene los atributos que se utilizaran para la clasificación de la información. Hablando en términos de eficiencia es mucho mejor el funcionamiento utilizando un wrapper debido a que se obtiene la información deseada del texto analizado pero cuesta más tiempo y trabajo desarrollar un algoritmo que permita el análisis del texto mientras que utilizando un filter es sencillo la implementación aunque no tendrá el resultado esperado[39].

La solución para iniciar con un análisis exhaustivo de la información es crear una herramienta que permita el análisis de la información textual de manera automática para obtener información relevante y así generar una estructura de los atributos que son principales en nuestra búsqueda.

La extracción de conocimientos es un proceso definido por una serie de pasos que son aplicados a los datos disponibles con una serie de parámetros que son útiles para la extracción de patrones útiles para la clasificación de los textos. Este proceso debe ser realizado de manera iterativa donde en las primeras ejecuciones debe obtener retroalimentación por parte del usuario para determinar la eficiencia del algoritmo convirtiéndose en primeras corridas con aprendizaje semi supervisado. La minería de textos se está volviendo en un área importante de investigación debido a la necesidad de obtener conocimiento de un gran número de documentos disponibles especialmente en la web. Minería de Textos y Minería de datos son las principales áreas que son dedicadas al análisis de información que es similar en algún sentido, las técnicas usadas en minería de datos pueden ser adoptadas para el área de minería de textos aunque la minería de datos trabaja con información estructurada y la minería de textos trabaja con información sin un orden predeterminado (No supervisado).

Actualmente XinChen y Yi-Fang presentaron una técnica de minería de textos que localiza una serie de reglas de asociación desde documentos hacia un usuario en particular, el sistema determina un perfil por cada usuario en base a los documentos que consulta de esta manera obtiene el conocimiento necesario para obtener las reglas semánticas. El trabajo propuesto resultó con un alto nivel de predicción[40].

BoyiXu presenta una técnica de minería de textos la cual establece un diseño e implementación de una página web de clasificación con el algoritmo CUCS. El algoritmo combina un método de

clustering sin supervisión y un método de máquinas de vector de soporte los cuales son implementados en el módulo de clasificación dentro de la página web la cual tiene un acercamiento para la clasificación del tipo de sitio al cual se tiene acceso clasificándolo según su contenido[41].

Shiqun Yin[41] presenta un algoritmo que reduce una página web mediante reglas de clasificación basándose en los atributos que presentan conforme se lee el sitio web, al estar utilizando el algoritmo de clasificación reduce y combina mediante reglas de extracción. El funcionamiento interno de este algoritmo genera una tabla con varios atributos los cuales cada uno tiene un atributo que determina el peso de la importancia de cada segmento que es leído de la página web. Este funcionamiento ayuda a reducir el número de características dentro del vector manteniendo el mismo esquema de clasificación. La precisión de este algoritmo es alta, la velocidad es media y las reglas de asociación son eficientes. Debido a que las reglas de asociación son basadas en el peso algunos detalles de la información se pierden, en algunos sitios web donde la información no es fácil de clasificar tiene un funcionamiento poco óptimo.

El proceso de minería descrito por RonenFeldman se enfoca hacia la exploración computarizada de un gran número de datos y mediante el análisis de grandes cantidades de información proceden a establecer patrones en el comportamiento de la información. El trabajo propuesto indica un acercamiento intermedio el cual es llamado minería de textos en términos base se trata de descubrir conocimiento enfocándose en una serie de palabras y frases que son extraídas directamente de un documento. Estos elementos son utilizados en los niveles altos de taxonomía y son usados para el descubrimiento de conocimiento que ayudara a generar la clasificación de la información revisada, esto explica un funcionamiento interno de lo que implica la implementación de minería de textos. El uso de minería de textos resulta importante para la búsqueda y entendimiento de la información contenida en los múltiples repositorios de artículos. El inicio de las definiciones y conceptualización sobre minería de texto y métodos utilizados para la extracción del conocimiento fue realizado por JanParalic Peter Bendar. Existen diferentes técnicas para la aplicación de minería de textos como reglas de asociación y modelos de clasificación son las principales técnicas que ayudan a la exploración de posibilidades detectadas en el texto analizado. El sistema Webocrat propuesto por el autor obtiene información de sitios web que revisan el flujo de datos dentro del sitio web desde discusiones en foros o chats, publicación de documentos, navegación en el sitio y despliegue de

la información; de esta manera generan una serie de preguntas de interés que ayudan a la obtención de información útil para el análisis utilizando servicios para ayudar al acceso de manera personal según las necesidades del usuario[40].

Las técnicas de minería de datos pueden ser utilizados en minería de textos como lo propone Helena Ahonen ella inicia con una relación de análisis de tareas aplicado en minería de datos relacionándolo con la clasificación de las frases extraídas del texto. Una vez establecida la relación genera un framework que sigue como objetivo principal el descubrimiento de conocimiento mediante procesamiento de la extracción de información siguiendo una serie de pasos. El primer paso para iniciar con la implementación de minería de datos es generar esquemas de información comunes que puedan llegar a presentarse y mediante esto establecer reglas de asociación que ayuden a la clasificación de la información. El trabajo de Helen Ahonen presenta un ejemplo donde obtiene información de un documento procesando el lenguaje natural contenido y obteniendo la clasificación esperada del documento. La aplicación fue demostrada utilizando experimentos con información real mostrando las reglas de asociación por segmentos de información entre distintos documentos ambos procesos fueron analizados según los resultados obtenidos, este proceso se basa en las reglas de asociación que se detectan dentro del documento analizado y describen una serie de pasos que incluyen limpieza de la información, estandarización de la información, reglas de asociación temporales y reglas de minería propuestas por KjetilNorvag . Las reglas propuestas consideran también pruebas sobre información real utilizada por cerca de 38 días con la página en línea de "The Financial Times" , la aplicación de las reglas en este tipo de información dio como resultado la extracción de términos, palabras base para detener el análisis así como frases que se consideran importantes para la clasificación. Las reglas de clasificación son evaluadas según los resultados obtenidos de esta manera se define qué tan óptima es la regla aplicada a la información.

Existen técnicas para descubrir las reglas de asociación en múltiples documentos, estas reglas fueron propuestas por Handy Mahgoud la técnica la llamó EARTH (ExtractedAssociation Rule from Text) esta técnica depende principalmente del análisis de las palabras clave extraídas del documento que son utilizadas para la implantación de las reglas de asociación, todo este proceso fue desarrollado utilizando C# y XML. Este trabajo se enfoca en aplicar la técnica en un inicio dentro del abstract de los documentos analizados aunque si se desea puede ser aplicado

el método a todo el documento, el proceso se basa en arboles de decisión en donde según el camino que se tome es el tipo de clasificador que se utiliza. El uso de árboles de decisión fue un trabajo elaborado por MohammadMasud Hasan y MofiuereRahman en este experimento se procesaron datos para encontrar relaciones entre palabras utilizando un algoritmo de a prioridad aplicando diferentes medidas en los objetos obtenidos. Los resultados mostraron que entre más grande sea el número de reglas de asociación mejores resultados se obtienen.

El uso de minería de textos para el apoyo en áreas diferentes a la de tecnologías de información está presente existe una aplicación llamada EVEX el cual se encarga de búsqueda de contenido relacionado a Biología, se utiliza de una manera constante esta aplicación donde busca en base al gen de la familia para extraer especies que se van a cruzar e intentar localizar candidatos. Este proceso inicia de manera manual evaluando la red resultante del análisis de los genes previos, esto para evitar tener elementos no necesarios en la implementación del proceso de minería de textos, acorde con la estructura utilizada en minería de textos corresponde a la preparación de los datos para continuar analizando la información con la aplicación mediante técnicas de minería de textos y expresiones relacionadas se identifica los elementos candidatos que pueden generar nuevas redes de genes, este trabajo busca la explotación del sistema EVEX para localizar la regularización del metabolismo utilizando el caso NADP(H)[42].

La información disponible en áreas como biología se ha ido incrementando en los últimos años de manera exponencial acorde a lo que nos presenta Martin Gerner mucha de la información que se agrega a estas librerías contiene información que es crítica, una gran parte de la minería de textos aplicada a estos documentos solo se enfoca a resúmenes o palabras clave lo cual deja que mucha información útil se pierda en esos filtros aplicados. El trabajo que presenta Martin Gerner es una recopilación que reúne técnicas de minería de textos donde extrae e integra la información obtenida de diversas herramientas cubriendo áreas como reconocimiento de entidades, eventos a nivel biomolecular, y contextualización de los resultados obtenidos por la herramienta. La aplicación tomo 10.9 millones de abstracts del sitio de MEDLINE y cerca de 234 000 artículos de libre acceso de la librería PubMed donde obtuvo cerca de 36 millones de referencias; toda esta información fue recabada por el investigador en una herramienta mediante técnicas de minería de datos este tipo de trabajos presentan un crecimiento del área de minería de textos en bioinformática[43].

2.5. MODELO DE INVESTIGACION.

El diseño de una arquitectura que permita un entendimiento apropiado del lenguaje natural en base a auto aprendizaje es complicado, por lo que se debe delimitar el área en que se va a trabajar y de esta manera poder delimitar el área de conocimiento. El uso de una técnica de minería de datos para mejorar el funcionamiento en la localización de información esto se puede implementar utilizando un algoritmo genético para la asignación de los clusters sobre los que se buscará la información solicitada.

La implementación de minería de datos para la solución de un problema implica la necesidad de implementar una metodología orientada a la implementación, donde existen diversas metodologías creadas de manera personalizada orientadas al tipo de atributos que van a revisarse, este tipo de metodologías son poco recomendables para nuestra implementación ya que necesitamos una metodología adaptativa para poder tener un comportamiento evolutivo.

Esto llevó a la empresa SAS, poner a disposición la metodología SEMMA[44] que por sus siglas en inglés (Sample, Explore, Modify, Model, Assess). Mientras que un grupo de Empresas de países europeos creó una metodología CRISP-DM por sus siglas en inglés (Cross- Industry Standard Processor Data Mining).

La metodología SAS se caracteriza principalmente por agregar prioridades a sus fases desde el punto de vista técnico, esto enfocándose en las prácticas usadas para su implementación y obtención de resultados. Esto implica un trabajo directo con una muestra de la población y va implementando directamente la manipulación de los datos, clasificación de variables e inmediatamente comienza con el análisis de los mismos. La metodología consta de las siguientes fases: Muestreo (simple), Exploración (Explore), Manipulación (Modify), Modelado (Model) y Valoración (Assess).

La metodología CRISP-DM[45] fue creada por un consorcio europeo conformado por Empresas de Dinamarca, Alemania, Inglaterra y Holanda que se unieron con el objetivo de crear una metodología de libre distribución que busca el cumplimiento de objetivos desde el punto de vista Empresarial, donde contiene un ciclo que comienza con la metodología.

La intención de fijar la metodología como libre distribución implica que se puede trabajar con cualquier herramienta para desarrollar el proyecto que esté disponible en el mercado aplicando así una característica adicional que es el de ser una metodología equitativa.

La elección para el diseño de la arquitectura será la de la metodología CRISP-DM, esto principalmente por ser una metodología libre que puede adaptarse al problema además de tener la libertad de modificarse según el problema que se esté trabajando.La metodología CRISP-DM se caracteriza por comenzar su análisis desde una perspectiva global enfatizando en el conocimiento del problema, de esta manera está más apegado al proyecto como metodología implementado en el diseño de la arquitectura. La metodología consta de seis fases que interactúan entre si según lo indica la siguiente figura:

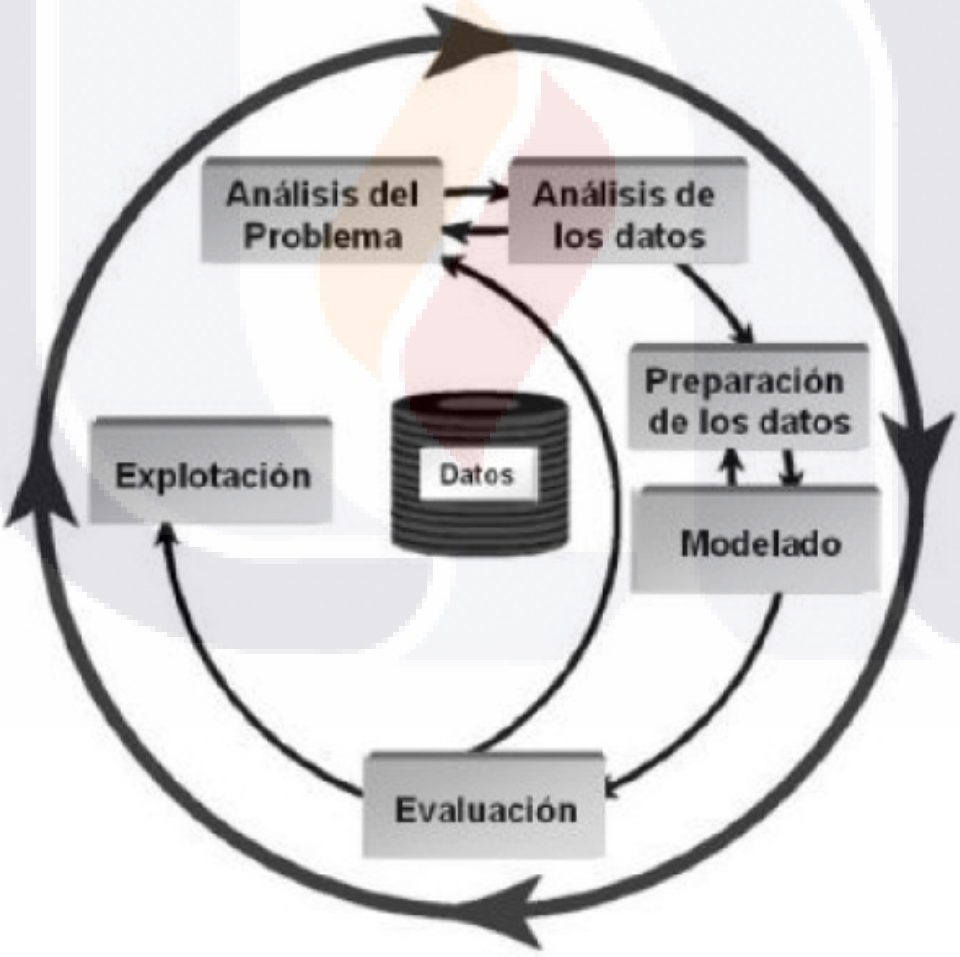

Figura 3. Fases en la Metodología CRISP-DM

La figura 3 muestra las fases en las cuales se trabaja para la adaptación constante del problema, donde inicia con un análisis del problema, una vez determinados los atributos importantes del problema se procede con el análisis de los datos, esta comunicación entre las primeras fases puede hacerse iterativo hasta tener un problema definido de una manera correcta y haber finalmente obtenido un análisis de los datos adecuado. El siguiente paso corresponde a la preparación de los datos, en este caso se asignan valores a los atributos que serán significativos para el análisis de la información y así obtener una optimización utilizando atributos importantes para la resolución del problema. La fase correspondiente al modelado se refiere al tipo de algoritmo que será utilizado para la implementación de Minería de Datos , una vez obtenidos los resultados se procede a la evaluación , en caso de no obtener los resultados esperados se puede volver a la fase del análisis del problema hasta obtener un resultado óptimo . Una vez obtenida una evaluación satisfactoria se procede con la explotación de los datos.

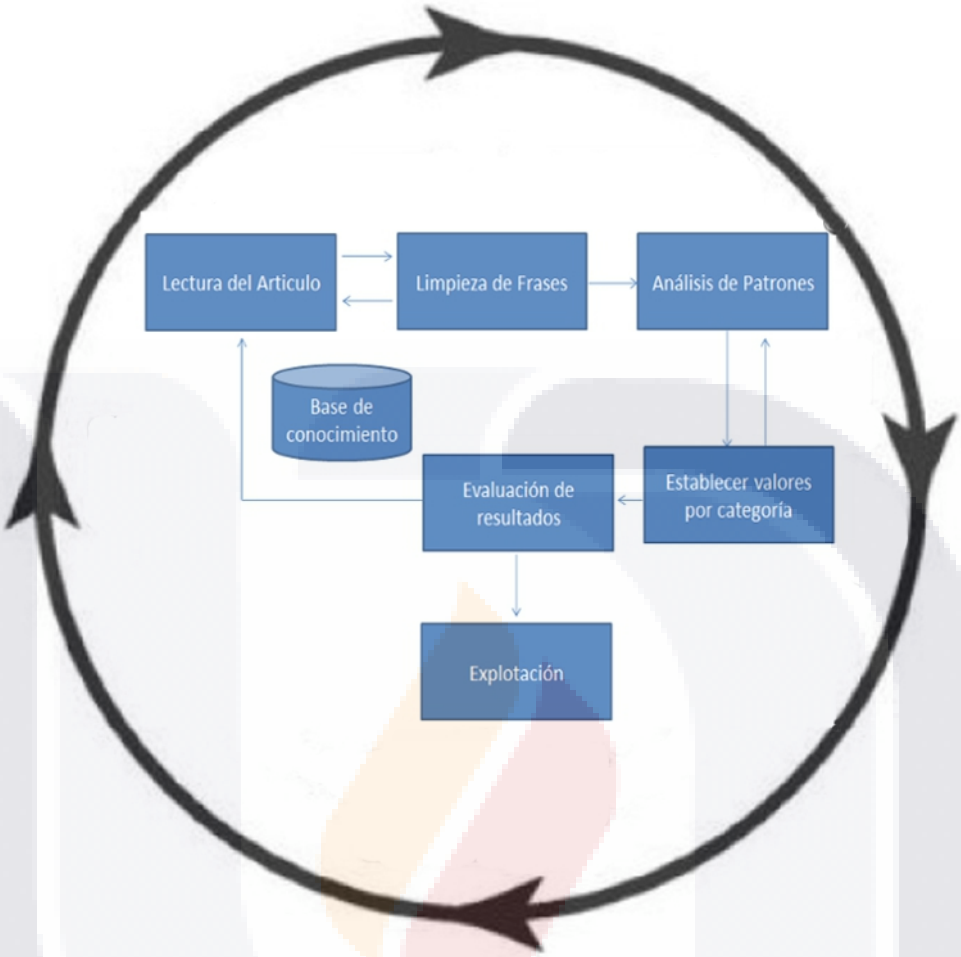

Figura 4. Metodología para la Clasificación y Búsqueda de Artículos de Investigación según Crisp-DM

La metodología (figura 4) para la clasificación y búsqueda de artículos de investigación nos permite utilizar la metodología CRISP-DM descrita por en la Figura 3, el proceso consiste en respetar la estructura propuesta por CRISP-DM donde se inicia con el análisis del problema, nuestro problema inicial es la lectura del artículo el cual puede venir en diferentes formatos y contener información variante desde imágenes, texto, caracteres especiales, etc. [46]

La figura 4 sigue la metodología CRISP-DM utilizando como base el funcionamiento recursivo hasta tener las frases consideradas importantes dentro del proceso de clasificación y búsqueda, en este caso la lectura del articulo corresponde al análisis del problema y a una limpieza de frases, estos dos procesos como se muestra en la figura 3, son recursivos para así poder tener

las frases que son importantes en la representación del contenido. El siguiente paso corresponde a la búsqueda de patrones el cual es el encargado de preparar los datos para poder iniciar el proceso de clasificación en el proceso relacionado a modelado correspondiente a establecer los valores por categoría, hay que tomar en cuenta que el proceso de modelado corresponde al módulo donde se encuentra el clasificador bayesiano o el wrapper. Una vez pasado por estos pasos se procede a evaluar los resultados, en caso de tener un resultado positivo se procede a la explotación de los valores y la alimentación de la base de conocimiento [46].

En base a la lectura del artículo nos lleva a la búsqueda de frases que no sean útiles para la clasificación del artículo el primer paso es detectar caracteres especiales y eliminarlos de las frases que no sean útiles para la clasificación. El siguiente paso consiste en detectar aquellas frases que sean repetidas en el artículo para así tener un arreglo de información útil para poder continuar analizando.

Una vez que tenemos nuestras frases únicas y limpias de caracteres especiales se procede a ejecutar un proceso que se encargue de la detección de patrones necesarios para localizar la categoría a la cual pueden llegar a pertenecer, este cálculo se realiza por cada frase única detectada y se le asigna una ponderación en base a el contenido actual que se tiene en la base de conocimiento.

El proceso continuo tomando los valores totales de la suma de aquellos resultados obtenidos por cada categoría en el proceso anterior, de esta manera podremos determinar en cual categoría es localizado. El siguiente paso es la evaluación de nuestros resultados, en este paso tenemos una base de conocimiento previa que se asignó con el mismo proceso pero con un número determinado de artículos por categoría, en caso de que los resultados no sean significativos para nuestra investigación se deberá proceder a iniciar desde el primer paso y volver a buscar otro método de clasificación y búsqueda. En cambio si todo el resultado del proceso resulta significativo se podrá asignar el artículo con sus frases y ponderaciones a la base de conocimiento o en su defecto determinar cuál es el resultado deseado de la búsqueda[47].

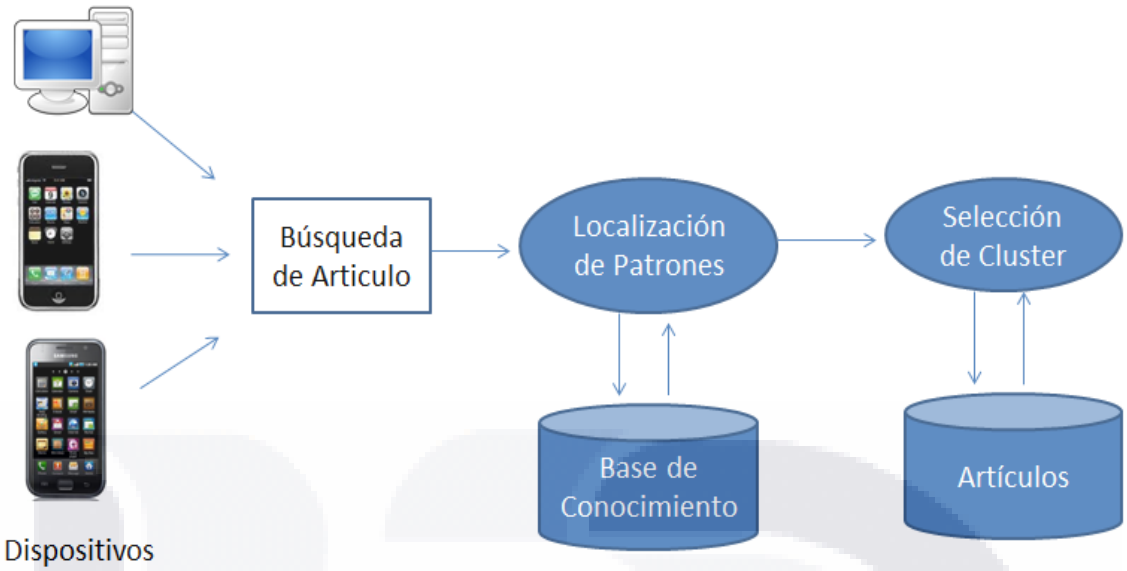

Figura 5. Arquitectura de Búsqueda

En la figura 5 muestra el funcionamiento interno de la arquitectura de búsqueda donde inicia una interface para introducir el texto con la información que se utilizará en el proceso de búsqueda de patrones dentro de una base de conocimientos para así poder obtener parámetros necesarios para la selección del cluster en donde se implementará la búsqueda del artículo, una vez logrado esto se realiza la búsqueda dentro de la base de datos en el proceso de "Localización de Artículos".

La figura 4 refleja los pasos que se siguen en la obtención de información mediante minería de textos , dichos pasos necesitan procesos fundamentales como es la obtención de patrones almacenados en el texto que se planea clasificar, así como un proceso de evaluación para poder explotar el conocimiento obtenido del texto clasificado como muestra en la figura 5.

Este esquema funciona utilizando como base la localización de patrones en el texto introducido este proceso se hace apoyándose en una base de conocimiento que es alimentada en una primera instancia de manera semiautomática con la información recabada de los artículos previamente almacenados.

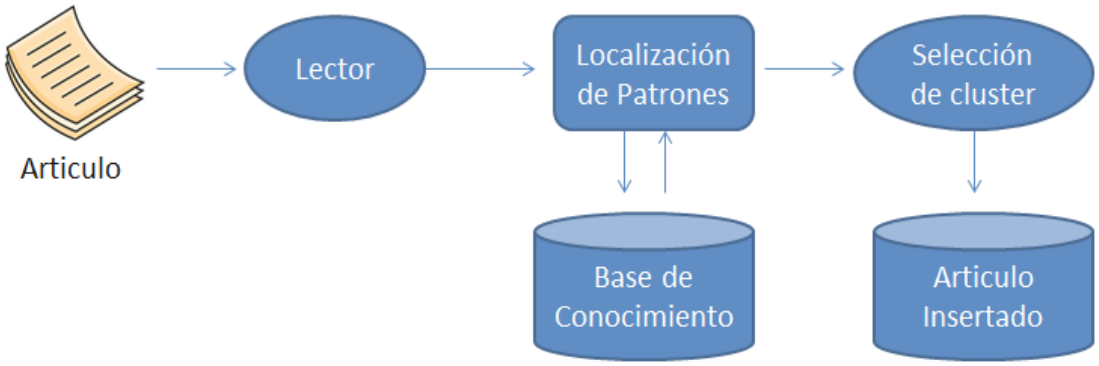

Figura 6. Proceso para dar de Alta un Artículo

La figura 6 muestra el proceso de alta de un artículo de investigación agregado por el investigador encargado de alimentar a nuestra base de datos, inicia con un lector que recorre toda el artículo de manera lineal buscando un patrón similar comparando con los patrones que hay en la base de conocimiento una vez que se localice en cual se relaciona se selecciona el cluster sobre el cual será dado de alta el artículo en la base de datos.

El uso de clusters o clustering implica la clasificación de diferentes grupos esto con el fin de generar grupos particionados que compartan alguna característica de esta manera se determina una medida de cercanía entre ellos creando grupos.

Este tipo de comportamiento utilizando clustering puede clasificarse como aprendizaje semiautomático, donde depende del tipo de algoritmo de clasificación que se implemente y el tipo de medidas utilizadas para alimentarlo, debido a que depende del tipo de información que se asigne ya que si se agregan un número indefinido de variables puede disminuir el rendimiento del algoritmo implementado. Esto es debido a que no todos los atributos son relevantes en la clasificación de información, se deben seleccionar aquellos que representen valores relevantes para lograr una correcta clasificación.

El establecimiento del número de clusters será en base al tipo de casos comunes que se encuentran en el área de lenguajes de programación debido a que es la principal base presentada en el problema de investigación teniendo los siguientes clusters:

- Instrucciones Condicionales if, case, switch.
- Ciclos, For, While , go to.
- Operaciones básicas suma, resta, multiplicación, división.

- Escritura: cout, system.out.println, writeln, printf.
- Conexión a base de datos, connect, mysqlconnect.

El algoritmo que se utilizará para el establecimiento de los clusters será mediante el algoritmo K-Means el cual será utilizado para enviarle los parámetros de clasificación de los artículos de investigación en este caso utilizaremos 5 clusters para lograr la implementación. El algoritmo funciona utilizando la siguiente ecuación:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2$$

Dadas un grupo de observaciones ($x_1,x_2,\ldots,x_n$) nosotros observamos una dimensión real del vector, el algoritmo k-means busca particionar un numero de n elementos en k conjuntos (k<= n) con $S=\{S_1,S_2,\ldots,S_k\}$ esto para minimizar la distancia entre los cuadrados de la ecuación.

Esta ecuación trabaja agrupando la información según el valor medio que represente la cercanía entre los elementos para de esta manera generar agrupaciones utilizando la variable indicada que ayudará a la clasificación de la información.

El algoritmo que se encargará de la clasificación de la información para poder determinar en que cluster es dado de alta, se establece consultando la base de conocimiento y comparando la información que está contenida en el artículo, mediante el siguiente algoritmo:

1. Apertura del archivo PDF.
2. Lectura de la línea
3. Comparar el contenido de la línea con la información que está en la base de conocimiento.
4. Comparar los valores y al final toma aquel que tenga el valor mayor de coincidencia.
5. Se suma al contador de la categoría en donde tiene mayor similitud.
6. Regreso al paso 2 hasta que termina el archivo

Estos son los 2 algoritmos que se utilizan para el funcionamiento del trabajo que presentamos en este artículo teniendo como base un algoritmo K-Means que utilizamos para la Clusterización y un algoritmo básico para la comparación de la información contenida en el

artículo de investigación que nos permitirá establecer a que cluster pertenece.

2.6DESCRIPCION DE CONSTRUCTOS, VARIABLES OPERACIONALES Y ESCALAS.

La escala utilizada para los valores será utilizando Likert en valores del 1 al 7 para así evitar sesgos y asegurar la calidad de la información extraída.

Constructos:

- En escala del 1 al 7 que tan preciso fue el resultado obtenido por el buscador de artículos de investigación.
- Que tan rápido considera el tiempo de respuesta considerando 7 como muy rápido.
- La interfaz de búsqueda considera que es ideal para un investigador, considerando el 7 como muy ideal.

2.7 DESCRIPCION DE LOS PRINCIPALES ESTUDIOS RELACIONADOS.

Existe una línea de investigación llevada por el Phd Julian Szymanski donde trata de categorizar documentos en base a la similitud que se tiene enfocado a metas específicas de conocimiento, donde en un inicio busca información relacionada a ciertas áreas importantes de conocimiento para generar agrupaciones, todo esto dentro de la información encontrada en los artículos de Wikipedia. Estas agrupaciones de resultados serán representadas en forma de clusters con información similar.

Dentro del trabajo de generar agrupamiento de información en base al texto que se encuentra guardado se encuentra el trabajo del Phd Mutlu Mete donde establece generar reglas de relación entre la información que se tiene disponible y eliminar reglas innecesarias en la relación de textos, su principal aportación es la de implementar asociaciones a través de colecciones de textos, con el principal objetivo de extraer información de textos.

2.8 ANALISIS DE CONTRIBUCIONES Y LIMITACIONES DE LOS PRINCIPALES ESTUDIOS RELACIONADOS.

Las publicaciones relacionadas a trabajos de otros autores destacan en la utilización de clasificación semiautomática donde prevalece la alimentación de una base de conocimientos previa pero con supervisión constante para continuar con el trabajo[31], También existen

trabajos relacionados a la extracción de conocimiento mediante la asociación de conceptos que generan reglas para poder utilizarlos en la clasificación de elementos[28], así mismo se puede implementar la Clusterizacion de texto mediante la detección de patrones.

Autor	**Ventajas**	**Desventajas**
[5]Phd Julian Szymanski	Implementación de un cluster espectral mediante los métodos de KVV, JNW y SM, con esto define los clusters mediante los patrones de líneas.	Solo ha sido implementado en Wikipedia aún no se porta hacia otros contenedores de artículos de investigación.
[32]Phd Mutlu Mete	Asociación de información a través de conceptos mediante significado de textos con los que generan reglas de asociación. Proponen el uso de palabras clave o bien multi-frases que describen información a buscar.	Probar otros algoritmos para extraer el conocimiento de bases de datos textuales.
[29]P. Ponmuthuramalingam , T. Devi	Aplicación de técnicas de clustering en información con lenguaje natural contenido.	No existe un algoritmo que determine el porcentaje que tiene el método de elegir a cual cluster corresponde la información clasificada.
[39]MS. K.Mugunthadevi M.Phil scholar, MRS. S.C. Punitha, Dr. M. Punithavalli	Selección de documentos utilizando minería de textos, muestra como el clustering puede adaptarse a la clasificación de documentos.	No muestra un caso de estudio técnico donde puedan verse los resultados o bien información estadística con el porcentaje de error del algoritmo implementado.
[41]Ms. Chhaya M. Meshram, Prof. Rahila Sheikh	Un estado del arte de las técnicas utilizadas en minería	Falta de resultados en la implementación, solo es

	de texto para la clasificación de información.	analizada de manera teórica.
[40]Ms. Vaishali Bhujade, Prof. N. J. Janwe , Prof S.W. Mohod	Estado del arte de la utilización de minería de textos con diferentes algoritmos.	Técnicas de clasificación de texto por filtering y wrappers, sin casos de uso.

Tabla 1. Trabajos Relacionados.

Capítulo 3 Marco Referencial.

3.1MODULO DE BASE DE CONOCIMIENTO.

El establecimiento de una base de conocimiento parte primero de indicar las áreas en las cuales tendrá información necesaria para auxiliar a la clasificación apoyándose en el concepto de aprendizaje semiautomático. Las áreas de interés para la clasificación son las siguientes:

- Programación.
- Sistemas Operativos.
- Base de Datos.

El procedimiento para diseñar la base de conocimiento consiste en obtener una muestra de 50 artículos por cada área en cada artículo se establecerá el siguiente proceso:

1. Se inicia un proceso de recorrido por todo el artículo palabra por palabra.
2. En un inicio se toman las palabras en un conjunto asegurándose que la palabra solo esté una vez.
3. Al finalizar el primer recorrido tendremos nuestro mapa inicial de las palabras que contiene el documento y procedemos a obtener los valores necesarios para la tabla de bayes.
4. El siguiente paso es establecer un recorrido y agregar a cada valor de nuestro conjunto el número de repeticiones que tiene en todo el documento.
5. Este proceso se tendrá que realizar por cada artículo.
6. Al tener los resultados de todos los artículos por cada categoría se genera la matriz de bayes donde tendrá el valor de repetición por cada palabra similar en los diferentes conjuntos.
7. Este promedio que tendrán por cada palabra será el valor que represente en la base de conocimientos.

El establecimiento de los valores en la base de conocimiento fue mediante bayes para así justificar las ponderaciones que se utilizarán en el clasificador.

keywords	bd	prog	so
articles fall	0.99999999	0	0
fountain	0.99999999	0	0
delivered	0.6236006	0.17230963	0.20408977
flood	1.00000002	0	0
data	0.99999999	0	0
executive it	0.99999999	0	0
estimated	0.71801339	0	0.2819866
amount	0.31998766	0.42128148	0.2587309
doubles	0.99999999	0	0
increases	0.40073108	0.24158756	0.35768133
faster	0.67072919	0	0.32927076
databases	0.88338395	0	0.11661609
although	0.36243064	0.30776175	0.32980756
small	0.45952244	0.23527262	0.30520496
dbase	0.99999999	0	0
business	0.59654171	0.13313425	0.27032403
activities	0.07506733	0.65960075	0.26533189
produces	0.51242375	0.29407133	0.19350485
stream	0.21170886	0.23399256	0.55429855
data	0.66898812	0.19260798	0.1384039
telephone	0.52482814	0.2175261	0.2576458
call	0.99999999	0	0
credit	0.87861338	0.12138662	0
card	0.35577418	0.29491624	0.34930949

medical	0.47679473	0	0.52320523
test	0.22304571	0.33896872	0.4379856
recorded	0.26706923	0.57560088	0.15732981
computer	0.99999999	0	0
rapidly	0.56983757	0.09447248	0.33568988
growing	0.99999999	0	0
national	0.43166124	0.25047556	0.3178632
aeronautics	0.99999999	0	0
space	0.52911701	0.13260224	0.33828073
administration	0.51249153	0.08496517	0.40254332
analyze	0.50458436	0	0.49541567
planned	0.27474689	0.4554986	0.26975453
expected	0.37505137	0.41452789	0.21042078
generate	0.45378551	0.29438762	0.25182687
terabyte	0.75342258	0	0.24657744
rate	0.36468506	0.50743694	0.12787801
second	0.35877651	0.34315963	0.29806385
pictures	0.1302808	0.48597868	0.38374049
generated	0.61384656	0.08480722	0.30134624
federally	0.99999999	0	0
funded	0.5355716	0.33296844	0.13145996
project	0.08194539	0.220766	0.69728863
store	0.27265921	0.12054332	0.6067975
thousands	0.26367131	0.21856828	0.51776041
bytes for	0.99999999	0	0
genetic	0.78625346	0.01303517	0.20071132
basescloser	0.99999999	0	0
lives	0.99999999	0	0
census	0.99999999	0	0

data			
bytes	0.18456205	0	0.81543793
encode	0.78350623	0.2164938	0
hidden	0.71383495	0.11094899	0.175216
describe	0.23688471	0.486234	0.27688134
raw	0.49743508	0.12687537	0.37568946
data	0.75342258	0	0.24657744
clearly	0.37623906	0.62376093	0
little	0.12001299	0.46757404	0.41241289
seen	0.15113677	0.40369177	0.44517147
if	0.31130518	0.20103084	0.48766395
analyzed	0.48333599	0.37708951	0.13957452
simple	0.25375899	0.41765453	0.3285865
statistical	0.70233812	0.14554943	0.15211236
techniques	0.6847733	0.11352749	0.20169911
analysis	0.46353695	0.28115514	0.25530792
developed	0.2536232	0.52290211	0.2234747
long ago	0.99999999	0	0
advanced	0.18777947	0.59433186	0.21788871
techniques	0.50093876	0.32297167	0.17608954
intelligent	0.19155426	0.77525627	0.03318945
data analysis	0.99999999	0	0
result	0.43421136	0.29449319	0.27129543
growing	0.35465905	0.08819755	0.55714338
gap	0.237695	0.29555314	0.46675181
between	0.32917108	0.39099388	0.27983506

generation and	0.99999999	0	0
understanding	0.19152849	0.71444727	0.09402413
time	0.50458436	0	0.49541567
expectation that	0.99999999	0	0
data	0.63058146	0.16911387	0.2003046
valuable	0.61686165	0.24063184	0.14250653
resource	0.11151616	0.03081348	0.8576703
advantage	0.99999999	0	0
computer	0.08453738	0.74748379	0.16797875
community	0.11803008	0.24460031	0.63736963
challenges	0.3731597	0.39368997	0.23315033
find the	0.70697806	0.29302198	0
knowledge	0.54565044	0.42577701	0.02857256
data	0.70388994	0.14587102	0.15023905
potential	0.3971241	0.28904746	0.31382842

Tabla 2. Algunas de las palabras utilizadas por el Clasificador Bayesiano.

3.2MODULO DE CLASIFICACIÓN.

El módulo que se encarga de clasificar toma como entrada el artículo en formato PDF , el primer paso que hace es buscar de manera interna la base de palabras que tiene el artículo tomando cada palabra y guardándolas en un conjunto validando que no se repitan.

Una vez obtenidas las palabras que contienen en el artículo se toman solamente la primera letra con la que inicia cada palabra almacenándolas en un conjunto validando que solo esté

contenida una vez cada letra. Al término de este proceso se toma el conjunto de letras y se realiza un filtro en nuestra base de conocimiento para extraer aquellas palabras que inicien con las letras que se tienen en el conjunto almacenándolas en un arreglo diferente.

El proceso final consiste en una comparación de la información que se tiene en nuestro arreglo proveniente de la base de conocimiento y nuestro arreglo de las palabras contenidas en el artículo se inicia un método comparativo palabra por palabra, cuando se tiene el parecido con la palabra se toma el valor y se multiplica por el porcentaje de similitud para finalmente sumarse al contador de la categoría donde se encuentra dicha palabra.

Al finalizar la comparación se tendrán tres contadores sobre la similitud de cada categoría y aquel valor que contenga un valor mayor es a donde se asignará el artículo.

3.3BASE DE CONOCIMIENTO UTILIZANDO BAYES.

Este módulo comprende la justificación del proceso utilizado para la creación de la base de conocimiento utilizando un clasificador bayesiano, donde se representa la información en forma de tablas con un valor asignado de manera probabilística de esta manera proveemos una forma compacta de representar el conocimiento y un método flexible de razonamiento.

El teorema de bayes[48] toma como base el supuesto que todos los eventos son independientes, por lo cual se toma como base el cálculo de la probabilidad de que todos los eventos sucedan y se almacena de manera independiente. Una vez obtenido el promedio de cada uno de los elementos en sus diferentes categorías (base de datos, programación y sistemas operativos) se toma la suma del total acumulado por cada categoría y se almacena de manera independiente para posteriormente utilizarla en el cálculo de sus probabilidades.

Frase	Base de Datos	Programación	Sistemas Operativos
Artículo	P | Frase	P |Frase	P | Frase
Programa	P | Frase	P | Frase	P | Frase
Total	1	1	1

Tabla 3. Promedio por Frases.

Una vez obtenidos estos resultados se determina que nuestra probabilidad de que el elemento se establezca en cada área es de 1/3 debido a que son 3 categorías con la misma probabilidad de que sea seleccionada, para establecer su criterio de probabilidad.

Frase	Base de Datos	Programación	Sistemas Operativos	Total
Artículo	P |Frase *P | D=BD / (P | Frase)* P | D=BD + P | Frase* P | D=Prog)+ P| Frase * P| D=SO)	P|Frase *P|D=Prog / (P | Frase)* P | D=BD + P | Frase* P | D=Prog)+ P| Frase * P| D=SO)	P|Frase*P|D=SO / (P | Frase)* P | D=BD + P | Frase* P | D=Prog)+ P| Frase * P| D=SO)	1
Programa	P| Frase*P| D=BD / (P | Frase)* P | D=BD + P | Frase* P | D=Prog)+ P| Frase * P| D=SO)	P | Frase*P| D=Prog / (P | Frase)* P | D=BD + P | Frase* P | D=Prog)+ P| Frase * P| D=SO)	P|Frase*P|D=SO / (P | Frase)* P | D=BD + P | Frase* P | D=Prog)+ P| Frase * P| D=SO)	1

Tabla 4. Obtención de Valores por Frase según la categoría.

Al tener la tabla con los valores calculados utilizando como base el valor equiprobable de que cada categoría suceda, se procede a utilizarlo dentro de nuestro clasificador bayesiano. El siguiente paso consiste en obtener las palabras que se tienen en el documento y si coincide con nuestro valor en la tabla final, se sitúa un 1 como valor para indicar que dicho valor se tomara en cuenta en las 3 categorías que contiene, este proceso se repetirá de manera constante hasta terminar de recorrer el documento para obtener finalmente una suma de valores probabilísticos donde el valor mayor según la categoría en la cual se situé será lo que determine el tipo de documento que tenemos.

CLASIFICADOR BAYESIANO				
PALABRA	**PRESENCIA**	**BD**	**PROG**	**SO**
Artículo	0	0	0	0
Programación	1	0.418448826	0.146563495	0.434987679
Desarrollo	0	0	0	0
PHP	1	0.158045186	0.420853371	0.421101443
Oracle	0	0	0	0

Java	1	0.676374608	0.255842751	0.067782642
Cluster	0	0	0	0
Row	0	0	0	0
Unix	1	0.214801002	0.42288897	0.362310028
Osx	0	0	0	0
TOTAL DE PALABRAS	**4**	**0.146766962**	**0.124614859**	**0.128618179**

Tabla 5. Valores Finales para el Clasificador Bayesiano.

Una vez obtenido el total de palabras en el ciclo, se obtiene el total de suma de probabilidades de cada categoría y según el valor que nos dé como mayor será el área en la cual deberá asignarse el documento.

La justificación para validar que nuestra base de conocimiento es significativa fue utilizando el método "Leave-one-out", el proceso utilizado fue el siguiente:

- Se almaceno la base de conocimiento en un "ArrayList"
- Se tomó el artículo que será dejado afuera, guardando primeramente todas sus frases en un conjunto, sin tomar elementos repetidos.
- Estos elementos se buscan en nuestro "ArrayList" que tiene toda la base de conocimiento, para dejar fuera los elementos que tenga en común con nuestro artículo de investigación.
- Una vez hecho este proceso se intenta clasificar dicho documento con los valores restantes de la base de conocimiento.
- Este proceso se realiza para todos los artículos que se utilizaron para formar la base de conocimiento de las 3 categorías.

El resultado obtenido al realizar este procedimiento ha sido el siguiente:

	Base de Datos	Programación	Sistemas Operativos
Valor	100%	100%	100%

Tabla 6. Comprobación del clasificador bayesiano "Leave-one-out".

Los resultados obtenidos pueden determinar que nuestra base de conocimientos es eficiente para clasificar cualquiera de nuestros artículos utilizados, según la validación “Leave-one-out”.

3.4MODULO DE BUSQUEDA.

La búsqueda consistirá en utilizar el motor utilizado para la clasificación, el proceso consistirá en tomar el texto que el usuario desea utilizar para su búsqueda, dividirlo mediante los espacios dejados por el usuario. Una vez obtenidos los segmentos de las palabras utilizadas se tomará primero las letras con las que inicia cada palabra almacenándolas en un conjunto evitando que se repitan para así continuar con una búsqueda dentro de la base de conocimiento.

Una vez obtenida la información de la base de conocimiento se hace una comparación de las palabras contenidas en el arreglo que introdujo el usuario contra las palabras obtenidas de la base de conocimiento, siguiendo el mismo procedimiento se obtiene un conteo de cada categoría, para así determinar en que cluster realizar la búsqueda, de esta manera el usuario obtiene una búsqueda con mayor nivel de precisión.

3.5 Clasificador utilizando un Wrapper

La utilización de un wrapper[49] como clasificador consta de un proceso que tiende a generar buenos resultados solo que su coste computacional es mayor, utiliza un algoritmo inductivo para establecer un ranking de subconjuntos de características similares, una vez establecido el subconjunto comprueba su precisión.

El algoritmo utilizado para el funcionamiento interno del wrapper es Ant System (AS), es el primer algoritmo desarrollado en el área ACO: Ant Colony Optimization. En esta se estudian sistemas artificiales que simulan colonia de hormigas reales, de donde se toma su inspiración. Estos sistemas son para resolver problemas de optimización combinatoria los cuales pueden ser descritos por problemas cuyo objetivo es encontrar la secuencia óptima de sus elementos.

El algoritmo Ant System es un algoritmo de búsqueda cooperativo inspirado por la conducta de las hormigas reales. La conducta de las colonias de hormigas es imitada por el algoritmo AS

usando agentes sencillos llamados hormigas (ants), que se comunican indirectamente por medio de un mecanismo inspirado en el rastro de feromona. Los rastros de feromona artificial, son un tipo de información numérica distribuida pero modificada por las hormigas y refleja su experiencia en la solución de un problema en particular.

El pseudocódigo del algoritmo AS es el siguiente:

Inicio
For t = 1 to Max_Iter do // Max_Iter es el número de iteraciones
For k = 1 to m do //m es número de hormigas (agentes)
Repetir *hasta que la hormiga k complete su recorrido*
Seleccionar *la siguiente ciudad que se va a visitar*
Calcular *la longitud del recorrido de la hormiga k*
Actualizar *los niveles de feromona*
Fin

La implementación del algoritmo clasificador wrapper utilizando AS, se basa en la agrupación de los elementos mayores localizados hasta el momento, tomando como base la base de conocimiento que se generó previamente por el clasificador bayesiano. Mediante esta técnica podremos obtener las palabras con mayor porcentaje de similitud entre nuestro documento a clasificar y la categoría a la cual pertenece en el documento.

3.6 Implementación de Clasificador Bayesiano

El proceso inicia tomando el artículo que se va a clasificar realiza una lectura completa del contenido del mismo obteniendo solamente las palabras que son únicas, una vez que se tienen las palabras únicas que son almacenadas en un arreglo, se continua a la obtención de posibles palabras que podrían coincidir de nuestra base de conocimiento con nuestro artículo de investigación.

El proceso inicia tomando las primeras letras de cada palabra y las almacena en un conjunto asegurándose que cada letra es única , una vez que se tiene nuestro arreglo con las letras iniciales únicas del contenido de nuestro artículo se procede a obtener un listado de todas las palabras que están en nuestra base de conocimiento con el siguiente código :

```
            Statement s = conn.createStatement ();
            s.executeQuery ("SELECT * FROM knowledgenewvalue WHERE keywords
like
                            '"+letra+"%'");
            ResultSet rs = s.getResultSet ();
            while (rs.next ())
              {
                String keyword = rs.getString ("keywords");
                Float area1 = rs.getFloat ("bd");
                Float area2 = rs.getFloat ("prog");
                Float area3 = rs.getFloat ("so");

                basecon.add(keyword);
                ratebasebd.add(area1);
                ratebaseprog.add(area2);
                ratebaseso.add(area3);


              }
```

Una vez que obtenemos nuestros arreglos que contienen por un lado las palabras únicas de nuestro artículo y en otro arreglo nuestras posibles palabras con las que puede coincidir de nuestra base de conocimiento procedemos a realizar el cálculo del porcentaje de similitud que puede existir para así determinar cuál es el valor que se tomara en cuenta para la categoría.

```
int caractsimilar = 0;
for (int e=0;e<basecon.get(d).toString().length();e++)
  {
   if ( e < texto.get(c).toString().length())
    {
     String temp1 = texto.get(c).toString().substring(e, e+1);
     String temp2 = basecon.get(d).toString().substring(e, e+1);
     if(temp1.equals(temp2))
       {
         caractsimilar++;
       }
    }
 }
float porcentaje = (float) 0.0;
porcentaje     =     (float)     ((float)     caractsimilar     /     (float)
basecon.get(d).toString().length());
```

El proceso inicia haciendo una comparación de bajo nivel entre las palabras que tenemos de nuestra base de conocimiento con nuestro artículo a clasificar, esta comparación se hace a nivel estructural revisando que la letra sea exactamente igual a la que tenemos en esa posición de nuestra base de conocimiento, una vez que se finaliza el proceso obtenemos el porcentaje con la cual es similar y tomamos aquel porcentaje que sea mayor obteniendo de esta manera la

palabra con la cual es similar de nuestra base de conocimiento así como el valor de cada categoría.

Los valores finales de cada categoría se van acumulando en diferentes contadores para finalmente obtener una sumatoria de todos los valores de cada categoría este proceso es el que determina a que categoría pertenece si es base de datos, programación o sistemas operativos.

3.7 Implementación de Wrapper

El proceso inicia tomando las primeras letras de cada palabra y las almacena en un conjunto asegurándose que cada letra es única , una vez que se tiene nuestro arreglo con las letras iniciales únicas del contenido de nuestro artículo se procede a obtener un listado de todas las palabras que están en nuestra base de conocimiento con el siguiente código :

```
            Statement s = conn.createStatement ();
            s.executeQuery ("SELECT * FROM knowledgenewvalue WHERE keywords
like
                        '"+letra+"%'");
            ResultSet rs = s.getResultSet ();
            while (rs.next ())
              {
                String keyword = rs.getString ("keywords");
                Float area1 = rs.getFloat ("bd");
                Float area2 = rs.getFloat ("prog");
                Float area3 = rs.getFloat ("so");

                basecon.add(keyword);
                ratebasebd.add(area1);
                ratebaseprog.add(area2);
                ratebaseso.add(area3);


              }
```

El proceso continuo de una manera similar al del clasificador bayesiano, es decir busca las palabras que tengan mayor parecido para así poder obtener el porcentaje de parecido y finalmente obtener aquellos valores que sean mayores según la palabra con la cual será relacionado dentro de nuestra base de conocimiento.

```
                    tempvalbd = Float.valueOf(ratebasebd.get(d).toString());
                    tempvalprog =
Float.valueOf(ratebaseprog.get(d).toString());
                    tempvalso = Float.valueOf(ratebaseso.get(d).toString());
                    if ((tempvalbd > tempvalprog)&&(tempvalbd > tempvalso))
                    {
                          tempval = tempvalbd;
                          tempcat = "bd";

                    }
                    if ((tempvalprog > tempvalbd)&&(tempvalprog > tempvalso))
                    {
                          tempval = tempvalprog;
                          tempcat = "prog";

                    }
                    if ((tempvalso > tempvalbd)&&(tempvalso > tempvalprog))
                    {
                          tempval = tempvalso;
                          tempcat = "so";

                    }
```

El siguiente paso consiste en obtener en un arreglo los primeros tres valores con mayor rango con su respectiva categoría, este proceso se hace recorriendo las palabras que se localizaron de manera inicial con su porcentaje de similitud y se compararon todas dentro de un ciclo, con el ciclo mostrado anteriormente. Una vez terminado este proceso se obtendrán los tres valores finales con mayor ponderación para iniciar el algoritmo de colonia de hormigas.

```
  for(int c=0;c<posiciones_grandes.size();c++)
    {
       int posi =(Integer) posiciones_grandes.get(c);
       float por_calculadown=(Float) porcentaje_tabla.get(posi) ;
        down +=  ((Math.pow(0.01, 1)) * (Math.pow(por_calculadown, 5))) ;
    }
```

El ordenamiento de las posiciones se hace mediante un ciclo donde se calcula el valor utilizando la función Math.pow, con el valor de la feromona y multiplicándolo por 5, valores propuestos para el cálculo del valor final que usaremos en nuestro siguiente segmento de código.

```
float suma=(float) 0.0 ;
for(int d=0;d<posiciones_grandes.size();d++)
  {
   int posi =(Integer) posiciones_grandes.get(d);
   float por_calcula=(Float) porcentaje_tabla.get(posi) ;
   float fero =Float.parseFloat(feromona_tabla.get(posi).toString()) ;
   float Pi =  (float) (((( Math.pow(fero, 1))*
                              (Math.pow(por_calcula, 5)))) / down);
```

```
   suma+=Pi;
multi_grandes.set(posi,Pi);}
```

El proceso anterior continúa con el cálculo del valor obtenido por el algoritmo de colonia de hormigas, este proceso nos lleva a obtener nuestros valores finales que usaremos en los recorridos planteados para la obtención de los resultados finales.

```
for(int d=0;d<posiciones_grandes.size();d++)
 {
       posi =Integer.parseInt(posiciones_grandes.get(d).toString()) ;
       por_calcula=Float.parseFloat(porcentaje_tabla.get(posi).toString())  ;
       fero =Float.parseFloat(feromona_tabla.get(posi).toString()) ;


      float Pi =  (float) (((( Math.pow(fero, 1))*
                      (Math.pow(por_calcula, 5)))) / down);

      suma+=Pi;
      multi_grandes.set(posi, Pi);
      lastpercent = Float.valueOf(porcentaje_tabla.get(posi).toString());
      lastcat = categoria_tabla.get(posi).toString();

    }
```

El código anterior nos da como resultado los valores finales que obtendremos de nuestro algoritmo de colonia de hormigas, de tal manera que podremos determinar donde es el lugar donde se clasificara el artículo que le enviamos a nuestro Wrapper.

3.8 Resultados

Las pruebas realizadas para la comprobación del comportamiento del clasificador bayesiano constan de tomar el 10% de cada categoría utilizada para nuestro buscador, pero los artículos a utilizar deberán ser diferentes de los artículos utilizados para el establecimiento de la base de conocimiento que se utilizó en los capítulos anteriores.

El propósito de la utilización de artículos diferentes a nuestra base de conocimiento es para poder establecer la calidad de nuestro clasificador bayesiano en un ambiente diferente al original.

En la investigación que se realizó dentro de nuestro marco teórico nos damos cuenta que el otro clasificador utilizado por la mayoría de la literatura es el llamado wrapper, lo cual es un clasificador basado en una metaheuristica.

La metaherisitca que vamos a utilizar es colonia de hormigas esto debido a su flexibilidad para resolver problemas, se tomó un valor de 0.01 para la feromona.

La base de conocimiento utilizada para correr el clasificador bayesiano como el wrapper es la misma, así como los artículos utilizados son los mismos para nuestro clasificador bayesiano y wrapper.

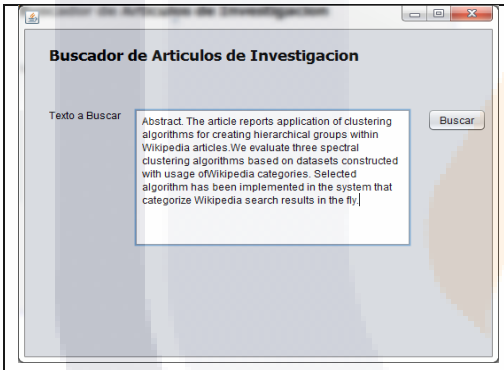

Figure 7.1 Interface de búsqueda para el Wrapper y el Clasificador Bayesiano

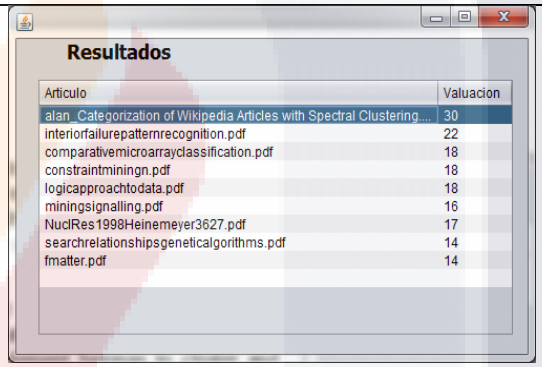

Figure 7.2.Resultados de la Busqueda.

La interface para la búsqueda en el clasificador bayesiano y en el Wrapper es la misma debido a que el proceso es interno, en el momento de la búsqueda como se muestra en la figura 7.1 donde se puede ver la interface que es utilizada para capturar el texto a localizar en nuestra base de conocimiento y finalmente obtenemos un listado de los artículos de investigación localizados con su respectiva puntuación.

Los resultados son los siguientes:

Wrapper:

Categoría	Aciertos
BD	2
Programación	8
S.O.	10
Total	20

Tabla 7. Aciertos de Wrapper.

El wrapper nos muestra un acierto absoluto en sistemas operativos, un 20% de error en programación y un 80% de error en la categoría de base de datos, la posible selección podría haber sido afectada debido a los recorridos y los cambios de la feromona. Además de la posible similitud entre las frases utilizadas para el área de base de datos y programación.

Clasificador Bayesiano:

Categoría	Aciertos
BD	10
Programación	10
S.O.	0
Total	20

Tabla 8. Aciertos de Bayesiano.

El clasificador bayesiano tiene un buen desempeño clasificando las categorías de base de datos y programación, pero muestra un desempeño bajo en el área de sistemas operativos, se debe

analizar la razón por la cual tiene este comportamiento nuestro clasificador bayesiano. Hay que tomar en cuenta que los artículos clasificados son completamente diferentes a los que tiene actualmente nuestra base de conocimiento.

Errores en Wrapper:

Categoría	Errores	Clasificación
BD	8	Programación
Programación	2	BD y S.O.
S.O.	0	
Total	10	

Tabla 9. Errores en Wrapper.

Los errores lanzados por el Wrapper nos muestran que puede existir una relación en los términos utilizados en la base de conocimiento entre el área de programación y el de base de datos, esto debido al número de errores reportados, de la misma manera en el área de programación se muestran dos errores uno hacia el área de base de datos y otro en el área de sistemas operativos. Este tipo de comportamiento nos puede ayudar a determinar que nuestra base de conocimiento aún podría mejorar o bien debería segmentarse de una manera más detallada para ayudar a la fácil clasificación y localización de los artículos.

Errores en Clasificador Bayesiano:

Categoría	Errores	Clasificación
BD	0	
Programación	0	

S.O.	10	BD
Total	10	

Tabla 10. Errores en Clasificador Bayesiano.

La tabla 9 nos muestra un comportamiento óptimo en las categorías de base de datos y programación, tomando en cuenta que el contenido de los artículos es fuera de los recursos usados para la creación de la base de conocimiento, por lo que el desempeño nos muestra un funcionamiento óptimo en esas áreas. El área que salió mal clasificada es la de Sistemas Operativos donde muestra una tendencia a clasificar los contenidos de una manera absoluta en el área de base de datos, lo cual nos puede ayudar a determinar que posiblemente existen términos que pueden ser significativos entre ambas áreas, tomando en cuenta el tipo de artículos utilizados para la creación de nuestra base de conocimiento.

Los resultados nos lanzan una igualdad en el número de aciertos que mostraron cada clasificador, pero los resultados a pesar de ser similares podemos ver que el wrapper tiene un rendimiento menos constante en las 3 categorías ya que solo obtuvo el 100% de aciertos en una sola área la cual es la de sistemas operativos , mientras que nuestro clasificador bayesiano tiene un 100% de eficiencia en dos categorías y tiene un mal desempeño en sistemas operativos al no lograr clasificar de manera correcta los artículos que se encuentran en dicha área.

3.9 Artículos Publicados

Los resultados publicados iniciaron con un trabajo que busca la localización de patrones en una conversación para así lograr relacionarlo hacia una búsqueda de objetos de aprendizaje[50] este trabajo inicio con una base de conocimiento enfocada al contenido de los objetos de aprendizaje y mediante frases se hacía una ponderación aproximada en base a preguntas para poder armar una búsqueda interna dentro del repositorio de objetos de aprendizaje y así darle al usuario los resultados que necesita.

El primer trabajo publicado en Conielecomp 2012 ayudo a tener los principios para la creación de un motor de búsqueda y clasificación de información utilizando filtros para determinar qué clase de búsqueda se va a hacer, el siguiente paso consistía en la utilización de un clasificador utilizando la metodología Crisp-DM, se utilizaron técnicas de clustering para dividir las tres categorías utilizadas así como un clasificador bayesiano para establecer a que categoría pertenece el artículo tanto en la búsqueda como en la clasificación del artículo a la base de conocimiento. Nuestra base de conocimiento fue analizada y justificada utilizando el método "leave-one-out" el cual nos dio resultados satisfactorios según las pruebas en las tres categorías que presentamos[51].

Autores	Artículo	Evento	Contribución
Edgar Alan Calvillo Moreno, Miguel Meza, Jaime Muñoz, Alejandro Padilla, Felipe Padilla, Francisco Alvarez.	Use of Chatterbot for Accessing Learning Objects on Mobile Devices With a Data Mining Search Engine	Conielecomp 2012 ,IEEE	Utilización de técnicas de minería de datos para la identificación de patrones dentro de una conversación utilizando un chatterbot.
Edgar Alan Calvillo Moreno, Alejandro Padilla, Jaime Muñoz, Julio Ponce, Jesualdo Fernandez.	Searching Research Papers Using Clustering and Text Mining	Conielecomp 2013 ,IEEE	Presentación del clasificador bayesiano utilizando minería de textos como base para la búsqueda y clasificación de artículos de investigación.

Tabla 11. Publicaciones.

CAPITULO 4 RESULTADOS

El presente trabajo muestra como inicio una arquitectura para la búsqueda y clasificación de artículos de investigación en las áreas de programación, sistemas operativos y base de datos. La arquitectura elaborada utiliza como base la metodología Crisp-DM, esta metodología es utilizada normalmente en proyectos de minería de datos. La arquitectura se creó utilizando la metodología que toma como base el obtener resultados previos a la obtención del modelo de arquitectura final. Este tipo de metodologías que ayudan a establecer una arquitectura en base a iteraciones ayudan a tener un modelo que genere una mejor calidad en los resultados.

La arquitectura es la mejor aportación de este trabajo de tesis debido a que se probó tanto con el método "leave-one-out" , así como tomando una serie de artículos que fueran totalmente diferentes a los que se tienen en la literatura utilizada para la creación de la base de conocimiento, los resultados lanzados en ambos mostraron un buen desempeño. Esta arquitectura puede ser tomada como base en otros trabajos relacionados a la clasificación y búsqueda de información contenida en lenguaje natural.

Las pruebas utilizando el módulo de búsqueda y clasificación fueron desarrolladas en Java con un esquema de base de datos implantado utilizando MySQL, los algoritmos fueron segmentados en clases para dejar el código disponible para otros proyectos con fines similares a los presentados en la arquitectura utilizada en este documento.

El presente trabajo tiene como limitante la funcionalidad de agregar palabras que no estén clasificadas en nuestra base de conocimiento, para ir incrementando el acervo de palabras contenidas en la base de datos utilizada para los algoritmos de clasificación y búsqueda. Este proceso podría ser contemplado como trabajo de nivel doctoral, donde iniciaría justo al fin de este trabajo tomando como base la arquitectura implementada y se agregaría código dentro del algoritmo encargado de búsqueda y clasificación. Este proceso debe funcionar de manera asíncrona al proceso actual existente de búsqueda y clasificación, podría ser utilizando hilos que permitan la ejecución asíncrona de los procesos , donde uno sigue el proceso normal de búsqueda y clasificación mientras que el otro hilo se encarga de realizar todo el proceso del clasificador bayesiano para determinar los pesos que corresponde a las palabras que no fueron encontradas en nuestra base de conocimiento de esta manera al final del proceso tendremos

nuestro artículo clasificado y aquellas palabras que no existan en la base de conocimiento tendrán las ponderaciones correspondientes dentro de nuestra base de conocimiento.

CONCLUSIONES Y TRABAJOS FUTUROS

El presente trabajo tomó como objetivo general la de realizar una aplicación utilizando como metodología Crisp-DM que permita la clasificación y búsqueda de artículos de investigación de manera óptima en tiempo y resultados. La eficiencia primero de la metodología Crisp-DM para utilizarse en diversos tipos de problemas demuestran cómo esta metodología es ideal para la implementación en distintos proyectos en el área de minería de datos[52] demuestran como las fases que tienen pueden llegar a ser adaptadas a diferentes situaciones que pueden encontrarse en la clasificación y búsqueda de información.

La metodología Crisp-DM en el área de minería de textos toma como base un esquema cíclico que en base a las diferentes iteraciones se va moldeando en primera instancia el análisis del problema en conjunto con la información que se recibe. Estos dos procesos ayudaron a en un primer enfoque a delimitar el problema que se va a trabajar debido a que si no se segmenta de manera correcta nuestra cantidad de datos a analizar podría ser muy amplia y tener poca contribución en el estudio de la tesis. Una vez obtenido el problema que se va a solucionar en nuestro caso la clasificación y búsqueda de artículos de investigación en las áreas de programación, sistemas operativos y base de datos, se procede a la preparación de los datos. El proceso que se sigue para la preparación de los datos consiste en llevar un filtro que ayude a eliminar información que no es necesaria para la clasificación y búsqueda de los artículos de investigación, en nuestro caso se procedió a eliminar imágenes, caracteres especiales y números que por su propio contenido son difíciles de clasificar. Al finalizar este proceso se continuo con el modelado, en este paso ya se procede a generar una arquitectura tomando como base el proceso que seguimos llevando la metodología Crisp-DM esto nos ayudó a generar una arquitectura que permitiera iteraciones, análisis del contenido, preparación de los datos así como el modelado para proceder a la evaluación de nuestros resultados obtenidos por cada artículo de investigación o búsqueda implementada.

Los pasos establecidos por nuestra arquitectura creada en base a Crisp-DM nos ayudaron a generar una base de conocimiento de las áreas de programación, base de datos y sistemas operativos. La creación de la base de conocimiento fue utilizando el esquema de aprendizaje semiautomático[53] esto significa que en un inicio buscamos una cantidad igual de artículos de cada categoría, revisamos dichos artículos que realmente pertenecieran a las áreas que

necesitábamos y se generó un script con el objetivo de en base al teorema de bayes establecer una clasificación de las frases más significativas de cada categoría, obteniendo de esta manera una base de conocimiento con las ponderaciones necesarias para poder realizar búsquedas y clasificación de artículos de investigación de las categorías seleccionadas. Este proceso no se quedó desde la primera iteración como versión final se buscaron alternativas para justificar que nuestra base de conocimientos era eficiente como lo determina uno de nuestros objetivos principales, para este proceso se tomó el método "leave-one-out" de los artículos seleccionados previamente para así determinar qué tan eficiente es nuestra base de conocimiento.

El proceso "leave-one-out" consistió en tomar cada uno de los artículos contenidos en nuestra base de conocimiento , eliminar aquellas frases relacionadas a dicho artículo y una vez que teníamos nuestra base de conocimiento limpia de cualquier contenido relacionado al artículo tomado se intentó clasificar dicho artículo y el resultado se comparaba finalmente con la categoría a la cual debía pertenecer. Esta prueba fue realizada para cada uno de los artículos que se seleccionaron para conformar nuestra base de conocimiento. El resultado fue un 100% de aciertos en cada una de los artículos tomados para clasificar, dando como conclusión que nuestra base de conocimientos tiene un nivel de eficiencia del 100% al intentar clasificarse a sí misma.

La obtención de los resultados fue gracias a la creación de un algoritmo que nos permitió recorrer el contenido del documento en formato pdf, para analizar el contenido con la base de conocimiento establecida previamente y así determinar a qué categoría pertenece. Este proceso fue utilizando primero un script que nos permite la eliminación de contenido que no es necesario para la clasificación del documento como son palabras repetidas, imágenes, caracteres raros y frases menores de 3 caracteres. Este algoritmo corresponde a una de las fases de la metodología Crisp-DM la cual es indica que se debe realizar una limpieza de los datos a clasificar.

Los resultados obtenidos del recorrido del algoritmo fue necesario aplicar una técnica de clusterización para así lograr determinar a qué categoría pertenecen, según el resultado es a donde se deben dirigir en este caso programación, sistemas operativos o base de datos. Este

algoritmo trabaja en conjunto con nuestro algoritmo de recorrido para optimizar recursos y obtener un resultado final sin la necesidad de implementar varias ejecuciones del script.

El algoritmo encargado de unir el recorrido del artículo de investigación trabaja en conjunto con nuestro clasificador bayesiano, el cual se encarga de obtener los valores de cada frase obtenida como resultado de un proceso que consiste en primero tomar las frases únicas arrojadas por nuestro script encargado de recorrer el documento. Una vez tenemos las frases únicas se procede a realizar un nuevo filtro el cual solo toma la primera letra de cada frase y se asegura de que cada primer letra sea única, estos procesos es mediante la teoría de conjuntos asegurándonos de esta manera que sean únicos y así poder continuar con nuestro trabajo. El proceso que sigue es el de la extracción de aquellas palabras que inicien con las letras únicas que tenemos almacenadas, de esta manera no se recorre toda la base de conocimiento, solamente aquellas palabras que tienen mayor probabilidad de existir en nuestro documento.

En este punto tenemos por una parte aquellas frases únicas de nuestro documento a clasificar y aquellas frases que pertenecen a nuestra base de conocimiento que tienen mayor probabilidad de relacionarse con nuestras frases, el siguiente proceso consiste en determinar por cada una de las frases de nuestra base de conocimiento cual es más parecida a nuestras frases contenidas en el documento a investigar, una vez que obtenemos la cifra que determina a que frase es la más similar, obtenemos el porcentaje de similitud entre la frase de la base de conocimiento y el de nuestro documento a clasificar y dicho valor lo usamos para obtener lo equivalente de los valores obtenidos por cada categoría, hay que recordar que nuestro clasificador bayesiano determina que una frase puede tener ponderaciones en las tres categorías.

El proceso explicado pertenece al de la clasificación de un documento, pero en caso de realizarse una búsqueda de un artículo de investigación el proceso es similar. Este proceso consiste en obtener una cadena de frases que son las que el usuario utilizará para localizar el artículo de investigación que desea, el proceso sigue la misma logística toma las frases únicas, realiza las ponderaciones mediante el clasificador bayesiano y cuando tiene los datos finales, se enfoca en mostrar solamente los artículos que el usuario necesita sin la necesidad de mostrar las tres categorías.

La hipótesis nula indica que no es factible el diseño e implantación de un Filter que permita la localización y clasificación de artículos de investigación con una calidad en los resultados, analizada en base a aprendizaje semiautomático utilizando como base la metodología Crisp-DM.

El presente trabajo muestra cómo es posible la implementación de un Filter, que nos permite la localización y clasificación de artículos de investigación utilizando un clasificador bayesiano el cual en base a la información obtenida de las frases únicas del artículo de investigación nos permite la clasificación y localización de artículos.

La arquitectura propuesta en este trabajo de tesis utiliza como base la metodología Crisp-DM la cual a pesar de ser una metodología orientada a la clasificación en áreas de minería de datos, es posible implementar dicha metodología tomando como base los pasos a seguir. El área de minería de textos posee campos similares acorde a lo localizado en este trabajo de tesis, por lo que solo fue necesario visualizar en primera instancia una arquitectura que funcionara de manera global y respetando las posibles iteraciones que sugiere la metodología Crisp-DM.

El siguiente paso fue dividir en dos segmentos principales nuestra arquitectura, uno donde es dedicado única y exclusivamente a la clasificación de los documentos de investigación, con el clasificador bayesiano y alimentando nuestra base de conocimiento. El otro segmento de la arquitectura se compone de tomar la frase capturada por nuestro usuario para así determinar qué es lo que se necesita mostrar en pantalla, según el texto introducido por el usuario determina si el artículo es del área de programación, sistemas operativos o base de datos.

Según los resultados propuestos podemos determinar que la Hipótesis Nula se rechaza debido a que se cumplieron tanto las metas como los objetivos propuestos así como se solucionó la problemática planteada al inicio.

Los resultados nos demuestran que tenemos un rendimiento similar tanto en la utilización de un wrapper como en la implementación de un clasificador bayesiano, aunque a nivel de complejidad en el procesamiento así como en tiempos es más rápido el clasificador bayesiano.

Se podría continuar con este trabajo para intentar adaptar al clasificador bayesiano para que obtenga conocimiento de manera automática mediante los artículos que va procesando en su clasificación para de esta manera incrementar la eficiencia de nuestro clasificador bayesiano.

El proceso para seguir con este trabajo de tesis consistiría en utilizar la base de conocimiento que se generó utilizando la metodología Crisp-DM, esta base de conocimiento tiene una validación utilizando el método "leave-one-out", se puede continuar buscando una combinación entre un wrapper que tome aquellas palabras que no se encuentran clasificadas previamente en la base de conocimiento, una vez tomadas estas palabras colocarlas en un conjunto que nos permita saber cuáles son las posibles nuevas frases que se van a someter a un aprendizaje completamente automático, este proceso no implicaría la misma complejidad computacional que se presenta al utilizar un wrapper debido a que solamente clasificara aquellas palabras que no existan en nuestra base de conocimiento, una vez obtenidas las clasificaciones previas se procede a insertar dichas palabras en la base de conocimiento. Este proceso puede aplicarse para nuevas palabras a agregar en las categorías existentes una vez que se haya agregado el número total de artículos con el fin de obtener nuevas palabras se puede volver a utilizar el método "leave-one-out" para la justificación de toda la base de conocimiento. En caso de que el resultado no sea óptimo se deberá ajustar el wrapper para revisar el funcionamiento interno del algoritmo aplicado y volver a realizar el proceso de alimentar la base de conocimiento y justificar con el método "leave-one-out".

El trabajo de tesis presenta un método con el cual puede buscar un motor robusto para la búsqueda y clasificación de artículos de investigación, puede tomarse como base para generar una base de conocimiento para otras áreas con el fin de tener artículos de investigación clasificados y facilitar la localización de información para el investigador.

Glosario de Términos.

Terminó	Definición
Minería de Datos	La minería de datos (es la etapa de análisis de "Knowledge Discovery in Databases" o KDD), es un campo de las ciencias de la computación referido al proceso que intenta descubrir patrones en grandes volúmenes de conjuntos de datos. Utiliza los métodos de la inteligencia artificial, aprendizaje automático, estadística y sistemas de bases de datos. El objetivo general del proceso de minería de datos consiste en extraer información de un conjunto de datos y transformarla en una estructura comprensible para su uso posterior. Además de la etapa de análisis en bruto, que involucra aspectos de bases de datos y gestión de datos, procesamiento de datos, el modelo y las consideraciones de inferencia, métricas de Intereses, consideraciones de la Teoría de la complejidad computacional, post-procesamiento de las estructuras descubiertas, la visualización y actualización en línea.
Minería de Textos	La Minería de Textos es una (otra) tecnología emergente cuyo objeto es la búsqueda de conocimiento en grandes colecciones de documentos no estructurados.
Algoritmo	En matemáticas, lógica, ciencias de la computación y disciplinas relacionadas, un algoritmo (del griego y latín, dixit algorithmus y este a su vez del matemático persa Al-Juarismi) es un conjunto preescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad. Dados un estado inicial y una entrada, siguiendo los pasos sucesivos se llega a un estado final y se obtiene una solución. Los algoritmos son el objeto de estudio de la algoritmia.
Metodología	La metodología hace referencia al conjunto de procedimientos racionales utilizados para alcanzar una gama de objetivos que rigen en una investigación científica, una exposición doctrinal o tareas que requieran

	habilidades, conocimientos o cuidados específicos. Alternativamente puede definirse la metodología como el estudio o elección de un método pertinente para un determinado objetivo.
KDD	Proceso de extracción de conocimiento, principalmente relacionado al descubrimiento de patrones en grandes cantidades de información.
CRISP-DM	La metodología CRISP-DM consiste en un conjunto de tareas descritas en cuatro niveles de abstracción: fase, tarea genérica, tarea especializada, e instancia de proceso, organizados de forma jerárquica en tareas que van desde el nivel más general hasta los casos más específicos.
Cluster	Un cluster es una técnica de minería de datos que permite agrupar un conjunto de elementos con características similares basándose en las diferencias que muestran los elementos entre ellos.
Wrapper	Algoritmo basado en alguna heurística o metaheuristica para la clasificación automática de texto.
Clasificador	La clasificación trata de encontrar las características que identifican a un grupo para ser clasificado dentro de cierta clase

REFERENCIAS.

[1] Lluís Codina, "¿Web 2.0, Web 3.0 o Web Semántica?: El impacto en los sistemas de información de la Web," *I Congreso Internacional de Ciberperiodismo y Web 2.0. Bilbao: Noviembre 2009*, vol. O'Reilly Media Web 2.0 Conference 2004.

[2] Pavan Kumar Mallapragada, Rong Jin, and Anil K. Jain, "Active Query Selection for Semi-supervised Clustering," *2008 IEEE*, vol. 978–1–4244–2175–6.

[3] Chakrabarti, S, *Mining The Web: Discovering knowledge from hypertext data*, 2003rd ed. Elsevier Science.

[4] Xiaomu Song and Guoliang Fan, "A study of supervised, semi-supervised and unsupervised multiscale bayesian image segmentation," *2002 IEEE*, vol. 0–7803–7523–8102.

[5] Julian Szymanski, "Categorization of Wikipedia articles with spectral clustering," *Polish Ministry of Science and Higher Education*, no. N519 432 338.

[6] José Guillermo Moreno Franco, "Contenido de las Imágenes en Artículos Médicos: Estado del Arte," *SEMINARIO DE INVESTIGACIÓN I - UNIVERSIDAD NACIONAL DE COLOMBIA.*, 2008.

[7] Juan Ángel, Rodolfo Bertone, and Ramón García-Martínez, "Modelo de Proceso de Operación para Proyectos de Explotación de Información," *CACIC 2010 - XVI CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN*.

[8] Nele Verbiest, Chris Cornelis, and Francisco Herrera, "Selección de Prototipos Basada en Conjuntos Rugosos Difusos," *Proceedings of XVI Congreso Español sobre Tecnologías y Lógica Fuzzy*, pp. 638–643, 2012.

[9] E. Fernández, H. Merlino, M. Ochoa, E. Diez, P. Britos, and R. García-Martínez, "GESTIÓN ASISTIDA DE DOCUMENTOS EN UNA METODOLOGÍA DE EXPLOTACIÓN DE INFORMACIÓN," *CACIC-2005*.

[10] Michael W. Berry and Malu Castellanos, "Survey of Text Mining: Clustering, Classification, and Retrieval, Second Edition," Springer, 2007.

[11] Jesus Serrano-Guerrero, Francisco P. Romero, Javier de la Mata, and Jose A. Olivas, "BUDI:Plataforma para la búsqueda difusa de información en repositorios de documentos," *XIV Congreso Español sobre Tecnologías y Lógica fuzzy*.

[12] P. L. N. José Angel Martínez Usero, “Agentes inteligentes en la búsqueda y recuperación de información.” PLANETA UOC, 2004.

[13] Pere Masip, “Busqueda de Informacion academica en Internet.” .

[14] Robert Neumayer, George Tsatsaronis, and Kjetil Nørvåg, “TRUMIT: A Tool to Support Large-Scale Mining of Text Association Rules,” *Proceedings of the 10th European Conference on Machine Learning and Knowledge Discovery in Databases - European Conference*, vol. 6913 of Lecture Notes in Computer Science,, pp. 646–649, 2011.

[15] Pilar María Moreno Jiménez, “Estrategias y mecanismos de búsqueda en la web invisible.” .

[16] Minsoo Lee, Yoonkyoung Lee, Boyeon Meang, and Okju Choi, “A CLUSTERING ALGORITHM USING PARTICLE SWARM OPTIMIZATION FOR DNA CHIP DATA ANALYSIS,” *ICUIMC-09, January 15-16, 2009, Suwon, S. Korea*, vol. ACM 978–1–60558–405–8..

[17] José C. Riquelme, Roberto Ruiz, and Karina Gilbert, “Minería de Datos: Conceptos y Tendencias,” *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial. No.29*.

[18] Cristóbal Romero Morales, Sebastián Ventura Soto, and Cesar Hervás Martínez, “Estado actual de la aplicación de la minería de datos a los sistemas de enseñanza basada en web,” *Taller Nacional de Minería de Datos y Aprendizaje, TAMIDA2005*.

[19] “Caso de exito mineria de datos.pdf.” .

[20] Ronen Feldman and James Sanger, “The Text Mining Handbook: Advanced Approaches to Analyzing Unstructured Data,” .

[21] Do-Jong KIM, Young woon Park, and Doong-Jo Park, “A Novel Validity Index for Determination of the Optimal Number of Clusters,” *IECIE TRANS INF AND SYST*, vol. E84-D, 2001.

[22] Kaushik Suresh, Sayan Ghosh, Swagatam Das, and Ajith Abraham, “Automatic Clustering with Multi-Objective Differential Evolution Algorithms.,” *XIV Congreso Español sobre Tecnologías y Lógica fuzzy*, 2008.

[23] Paloma Moreda Pozo, “Los roles semánticos en la tecnología del lenguaje humano: anotación y aplicación,” .

[24] DANIEL T. LAROSE, *DISCOVERING KNOWLEDGE IN DATA*, vol. Published by John Wiley & Sons, Inc., Hoboken, New Jersey. .

[25] I. Shanthi and M. L. Valarmathi, “Comparison of Fuzzy C-Mean Clustering and K-Means Clustering for SAR Image Despeckling using Edge Detection,” *European Journal of Scientific Research*.

[26] M. Dash and H. Liu, "Feature Selection for Classification," *1997 Elsevier Science B.V*.

[27] Margarita Reyes-Sierra and Carlos A. Coello Coello, "Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art," *International Journal of Computational Intelligence Research*, vol. 2, No. 3, no. 2006, pp. 287–308.

[28] Julian Szymanski and Włodzisław Duch, "Representation of hypertext documents based on terms, links and text compressibility," *ICONIP 2010*.

[29] P. Ponmuthuramalingam and T. Devi, "Effective Term Based Text Clustering Algorithms," *International Journal on Computer Science and Engineering*.

[30] Julian Szymanski, "Towards Automatic Classification of Wikipedia Content," *IDEAL 2010*.

[31] Julian Szymanski, "Self–Organizing Map representation for clustering Wikipedia search results," *ICONIP (1) 2012*.

[32] Mutlu Mete, Nurcan Yuruk, Xiaowei Xu, and Daniel Berleant, "Knowledge Discovery in Textual Databases: A Concept-Association Mining Approach," *Springer Data Engineering*, pp. 225–243.

[33] Manuel Montes-y-Gómez, "Minería de texto: Un nuevo reto computacional."

[34] Marti Hearst, "What Is Text Mining?," *SIMS,UC Berkeley*.

[35] Anni R. Coden and EricW. Brown, "Automatic search from streaming data."

[36] "INFORMATION RETRIEVAL AND TEXT MINING," .

[37] Ah-Hwee Tan, "Text Mining: The state of the art and the challenges."

[38] K. Prasanna, M. Sankara Prasanna Kumar, and G. Surya Narayana, "A Novel Benchmark K-Means Clustering on Continuous Data," *International Journal on Computer Science and Engineering (IJCSE)*.

[39] MS. K.Mugunthadevi, MRS. S.C. Punitha, and Dr..M. Punithavalli, "Survey on Feature Selection in Document Clustering," *International Journal on Computer Science and Engineering (IJCSE)*.

[40] Ms. Vaishali Bhujade, Prof. N. J. Janwe, Prof S.W.Mohod, B.D.C.O.E. Sevagram, R.G.C.E.R.T. Chandrapur, and B.D.C.O.E. Sevagram, "KNOWLEDGE DISCOVERY IN TEXT MINING: A REVIEW," *International Journal on Computer Science and Engineering (IJCSE)*.

[41] Ms. Chhaya M.Meshram, Prof. Rahila Sheikh, 1B.D.C.O.E. Sevagram, and R.G.C.E.R.T. Chandrapur, "WEB TEXT MINING USING CLASSIFICATION ALGORITHM: A REVIEW," *International Journal of Engineering Science and Technology (IJEST)*.

[42] Suwisa Kaewphan, Sanna Kreula, Sofie Van Landeghem, Yves Van de Peer, Patrik R. Jones, and Filip Ginter, "Integrating Large-Scale Text Mining and Co-Expression Networks: Targeting NADP(H) Metabolism in E. coli with Event Extraction."

[43] Martin Gerner, Farzaneh Sarafraz, Casey M. Bergman, and Goran Nenadic, "BioContext: an integrated text mining system for large-scale extraction and contextualization of biomolecular events," *Published by Oxford University Press.*, vol. 2012.

[44] Rodríguez Montequín, M[a] Teresa; Álvarez Cabal, J. Valeriano; Mesa Fernández, and José Manuel; González Valdés, Adolfo, "METODOLOGÍAS PARA LA REALIZACIÓN DE PROYECTOS DE DATA MINING."

[45] Britos, P, Fernández, E, Ochoa, M, Merlino, H, Diez, E, and García Martínez, R, "METODOLOGÍA DE SELECCIÓN DE HERRAMIENTAS DE EXPLOTACION DE DATOS," *Workshop de Ingeniería del Software y Bases de Datos*, vol. XI Congreso Argentino de Ciencias de la Computación (2005).

[46] Chapman, P.: Clinton, J.: Kerber, R.: Khabaza, T.: Reinartz, T.: Shearer, C.: Wirth, R, "CRISP-DM 1.0 Step-by-step data mining guide, 1999.," .

[47] "Metodologia de Aplicacion del Data Mining," .

[48] Itzama Lopez Yañez, "Clasificador Automatico de Alto Desempeño."

[49] Nicholas Kushmerick, "Wrapper introduction for information extraction." .

[50] E. Alan Calvillo, Miguel Meza, Jaime Muñoz, Alejandro Padilla, and Felipe Padilla, "Use of Chatterbot for Accessing Learning Objects on Mobile Devices With a Data Mining Search Engine," *Conielecomp 2012*.

[51] E. Alan Calvillo, Alejandro Padilla, Jaime Muñoz, Julio Ponce, and Jesualdo Fernandez, "Searching Research Papers Using Clustering and Text Mining," *Conielecomp 2013*.

[52] Ing. Juan Miguel Moine, Dra. Ana Silvia Haedo, and Dra. Silvia Gordillo, "Estudio comparativo de metodologías para minería de datos."

[53] I. Olmos-Pineda and J. A. Gonzalez-Bernal, "Minería de Datos."

Anexos

Searching Research Papers Using Clustering and Text Mining

E. Alan Calvillo[1],Alejandro Padilla[1],Jaime Muñoz[1]
Centro de Ciencias Basicas[1]
Universidad Autónoma de Aguascalientes[1]
Aguascalientes, Mexico[1]
bsdemonio,jmauaa@gmail.com,apadilla@correo.uaa.mx

Julio Ponce[1], Jesualdo T. Fernandez[2]
Depto. de Informática y Sistemas[2]
Universidad de Murcia[2]
Murcia, España[2]
julk_cpg@hotmail.com,jfernand@um.es

***Abstract*—The time spent by users are almost two or more hours looking for papers that generates the possibility to make a search engine to optimize and precision in the results. This works purposes a better classification of research papers, the architecture works with a database of knowledge related with the topics of programming, databases and operating systems. That's the initial work of a classification using text mining techniques to search into the documents with natural language contained and get the best words of their content to get a database knowledge, that's the first step to get the desired knowledge also the proposed work use the same engine to make searches classifying the information introduced by the final user and searching in the correct cluster**

Keywords— text mining, clusterk-means,dabase,pattern,knowledge.

INTRODUCTION

The use of search engines to locate information has grown steadily based on the needs of users generating a snowball effect, where all the information is available in different websites, including information that is not useful or significant but also included information of scientific interest to remember that not all the information is by default in specialized research journals, transactions, letters and magazine in the area of computers science or advances in technology like Springer, IEEE, ACM or papers recognized of research and outreach.

The generation of multiple research papers goes to a generation of multiple repositories Springer, IEEE, ACM or papers recognized of research and outreach. This kind of segmentation of information makes a generation of hours of search papers when a researcher it's searching for a specific topic. The main goal to minimize the time inverted in searches and also makes best searches and has a minimal time of search [1].

There are currently functioning as search sites called metasearch[2] that help review the information that throw the other search engines[3], but searching normally, only responsible for presenting results on screen as a search interface running on multiple search engines.

The implementation of academic search engines have been evolving as the needs of the research sector mainly followed by Scirus Google Academic, science research, in general the performance of academic search engines depend on the results given by the commercial search engines[4] to finally use his own generation of indexes which store information for papers and primarily on the information contained in the databases of commercial search engines, where if it is possible to view the research paper and should not be available to send the user site where the user need purchase the paper. This kind of situations has been reviewed subsequently posing a smart search engine based on the use of ontologies and classification of information to conduct an information search through different search engines via the web [5].

The clustering is an unsupervised classification of patterns into groups the clustering problem has been addressed in many contexts and by result in many disciplines; this reflects its broad appeal as one of the steps in exploratory data analysis. However, clustering is difficult problem combinatory and differences in assumptions and contexts [15].

The implementation of a better tool to search research articles, it would be useful to the result and minimize times of search and also make a best

engine to get best results in every search of research papers.

The use of K-Means algorithm allow us to implement semi-supervised learning clusters using an algorithm so as to help identify approximate the text to search using predefined patterns and the implementation of a cluster algorithm [6] for consultations within the database manager MySQL (database manager that allows free use of multithreading, multi-search and multi-user) in order to obtain scientific research papers.

This paper is organized as follow: in section 2 the paper give an introduction of the problem outline presented in the search of research papers also in section 3 a solution to the problem outline with architecture to classify and locate research papers. In section 4 gives a case of study where show how the architecture works in a web environment. Finally, section five shows a benchmark of related papers.

PROBLEM OUTLINE

The users spend a lot of hours searching in the repositories of papers on topics related to the area of information technology and development, which requires the establishment of a search engine to locate items of research[5] in the area of programming languages allowing identification of basic patterns in the input text and the implementation of a data mining algorithm to help decrease the response time in the search within the database(MySQL) for locating scientific articles and as a prototype-level implementation that allows access from a browser(v.gr. Firefox).

The large hours spent by user makes necessary to develop a prototype to enable analysis of the performance for testing based in the amount of accurate results related to the type of search performed as well as the relationship obtained in articles. Since the obtained knowledge base must be taken as a starting point to determinate patterns within a word captured and to deduce the weight to be given by the user submit this information to the data mining algorithm. The main problem is to solve the next points:

- Develop architecture for the management and use of knowledge base adapting a correct interpretation of patterns related to each sentence.
- Implement an architecture using a mysql server and create database knowledge
- Develop a Filter to generate a classification of research papers and also searches
- Develop a Wrapper to generate a classification of research papers and also searches
- The time searching papers takes between two and four hours to locate the correct paper. The search engine must have a better time than the actual.

CONTRIBUTION

The implementation of data mining to solve a problem involves the need to implement a methodology focus into the analysis of pattern into the texts, where there are several methodologies custom built-oriented type of attributes that will be reviewed, such methodologies are not recommended for our implementation as the proposed work need a methodology to be adaptive evolutionary behavior

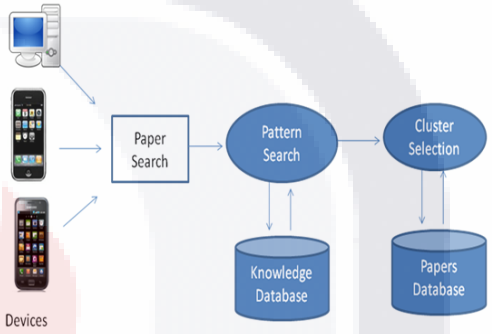

Figure 1.Search Architecture Model

In figure 1 shows the search architecture model, the main purpose is the use of text mining and also the first part of the architecture needed in our problem outline where the userbegins a search interface for entering text within the information used in the process of searching patterns within a knowledge base in order to obtain parameters for the selection of cluster where the search was implemented, once achieved the search is conducted within the database in the process of localization papers.

This current architecture works using as input the location of the research to get text patterns, that work reading line by line and in this process is supported by a knowledge base that is fed into a first semi-automatically with the information collected from items previously stored.

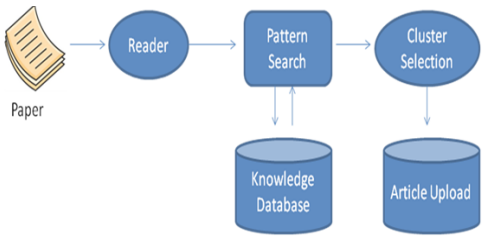

Figure 2.Pattern matching to store article.

The figure 2 uses the search pattern into the research paper looking to the database searching patterns of knowledge, with this the proposed work generate our engine to make pattern matching needed in the problem outline start with a pattern matching that read the article searching a similar pattern compared with the knowledge base once they locate in which it relates is selected cluster on which will be uploaded from the article in the database.

The use of clusters involves the classification of different groups partitioned that share a characteristic in this way determines a measure of characteristics between the stored information in the knowledge database.

This type of behavior can be classified as clustering semi-automatic learning, which depends on the classification algorithm, is implemented and the type of measures used to feed because it depends on the type of information is assigned and that adding a number undefined variables can degrade the performance of the implemented algorithm. This is because not all attributes are relevant to the classification of information. The process should select those that represent relevant values for proper classification.

The algorithm used for the establishment of clusters will be using the K-Means algorithm which will be used to send the parameters for classification of research papers in this case the search engine will use five clusters to achieve implementation. The algorithm works by using the following equation:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2$$

The formula represents a given set of observations (X1,X2… Xn) where each observation represent an element of the cluster with a d-dimensional real vector , k-means clustering aims to partition and the n observations into k sets (k <= n) S = { S1, S2, ….. Sk) that's to minimize the cluster where μi is the mean of points in Si.

The equation of k-means works by grouping the information according to the average value that represents the proximity between the elements to thereby generate clusters using the indicated variable to assist in the categorization of information

The implementation of the algorithm using K-Means is input first two parameters that determine the weight. The process have to establish the element belongs to cluster and eventually a label that represents the element, the development was done using Java when running the program shows the result in figure 3.

----------Cluster0---------
Qsort[3.0,32.0]
Push and Pop[3.0,22.0]
----------Cluster1---------
Oracle[19.0,20.0]
MySQL[8.0,10.0]
----------Cluster2---------
PHP[3.0,2.0]
Java[1.0,3.0]

Figure 3.Example of clusters using k-means.

The K-Means algorithm is responsible for the establishment of clusters, but the categorization of information to determine which cluster is selected is set by querying the knowledge base and comparing the information that is contained in the article using the algorithm in figure 4, with this the application can obtain our main objective to use a filter proposed as problematic in the third point.

1. *Opening PDF file*
2. *Read PDF line*
3. *Compare the contents of the line with the information that is in the knowledge base*
4. *Comparing whether there is a similarity of at least 80% between the line and the value of the knowledge base.*
5. *According to the result the Cluster is assigned*
6. *Return to step 2 until the file ends*

Figure 4.Algorithm of search patterns

These are two algorithms used for the operation of the work presented in this paper the basis of a K-Means algorithm that uses for clustering and a basic algorithm for the comparison of the information contained in the research article that will allow us set to cluster belongs.

CASE OF STUDY

The implementation of the search engine using data mining scheme works by using a web search to research this for easy portability to multiple platforms helping a simple web search interface where the user enter the information that want and finally showing a list of papers available.

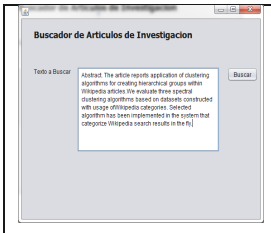

Figure 5.1User Interface of

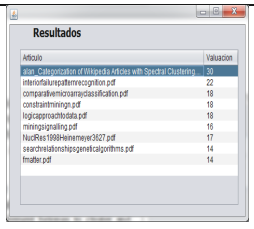

Figure 5.2.Results of the

the System Search	Search.

The Figure 5.1 shows the interface to be used for searching and browsing of research papers with a simple structure that helps the user to identify a single text area where the user enter text on the search and finally at the bottom were generated the results of this search. The time in the search takes one minute to search in all the database knowledge, making a best time than the usual when the researcher are searching in the web. The figure 5.2 shows results of the search in the second column of results, with their own value of comparison using the full text in the search engine and compared with their own category in this case data base.

1. *The item is placed randomly in each cluster*
2. *Compare the items without classification*
3. *Items comparative review of their distance from each other using the mean of each element*
4. *If it is near the item is added to the cluster, if not so return to step two*
5. *Once the cycle are finished the elements clustered*

Figure 6.K-Means algorithm implemented

The algorithm of figure 6 shows the steps used to implement the k-means algorithm where first assign an item to each cluster at random to start with the categorization using the support arrangements for the comparison and thus be systematically ordering the clusters which will make the search query.

RELATED WORK

The problem outline in this paper has been discussed previously by other papers; the present work attend first assigning weights to attributes on the other hand while working with the implementation of the creation of clusters as seen in Table 1.

TABLE 1.RELATED WORKS.

Author	Contributions	Limitations
Julian Szymanski[17]	The implementation of a cluster by spectral methods KW, JNW and SM with clusters defined by this line of patterns	It has only been implemented in wikipedia still does not behave to other containers research articles.
Mutlu Mete[18]	The association of information through concepts by means of texts that generate association rules. The use keywords or phrases that describe information.	Try other algorithms to extract knowledge from text databases.

The table 1 shows two works of different results, in their papers one it's using with Wikipedia on Poland, that work [17] propose access to classify information to an application to make the process. The proposed work in [18] is looking to create an auto generator of knowledge to classify the information both are working with a topic similar of this paper but they're using another techniques and not text mining.

CONCLUSIONS

This paper evaluates a way to optimize the information to be located within a structured framework with an initial knowledge base that helps the easy categorization of information by implementing a clustering for fast search and location as well as a textual analysis entered by the user as a basis for consultation, as future work is to implement an automatic learning that allows the steady increase in the manipulated texts. That kind of techniques allows making a best search engine using database knowledge to work with filter, wrapper or even ontology.

The use of text mining technologies are not used in web search or meta search, that kind of tools usually use only meta crawler to classify the information the current work shows how the search engine can be used and it should make a benchmark between the filter, wrapper and ontology to the next work.

As future work is consider to extend the search engine into another kind of devices using Android or IOS that's to generate a portable application to make searches in different kind of devices.

REFERENCES

Garrudo, Á.M.A., METABUSCADOR O-QE Junio, 2006

Navarra, P.L. and J.A.M. Usero, Agentes Inteligentes en la busqueda y recuperacion de la informacion, in Planeta UOC, S.L. 2004 Barcelona..

Codina, L., ¿Web 2.0, Web 3.0 o Web Semantica ? El impacto en los Sistemas de Informacion Web. O'Reilly Media Web 2.0 Conference 2004, 2009.

Masip, P. Busqueda de Informacion academica en Internet. 2009; Available from: http://www.slideshare.net/p.masip/buscadores-acadmicos-3052335 last consult 20/02/2012.

J. A. Olivas, J.d.l.M., J. Serrano-Guerrero, P. J. Garces, F. P. Romero, Desarrollo de Motores Inteligentes de busqueda en Internet en el Marco del grupo de Investigacion SMILe-ORETO. 2006. ISBN 84-9750-525-5.

Chakrabarti, S., Mining The Web: Discovering knowledge from hypertext data, Part 2. 2003.

M. Lee, Y.L., B. Meang, O.Choi, A Clustering Algorithm Using Particle Swarm Optimization For DNA Chip Data Analysis. 2009.

M. Dash , H.L., Feature Selection for Classification. Intelligent Data Analysis, 1997.

Szymanski, J., Categorization of Wikipedia Articles with Spectral Clustering. Spring-Verlag Berlin Heidelberg, 2011. 108-115-2011.

Coello, M.R.-S.a.C.A.C., Multi-Objective PArticle Swarm Optimizers : A survey of State of the art.

Do-Jong K., Y.W.P., D. Park, A Novel Validity Index for Determination of the Optimal Number of Clusters. 2001.

Duch, J.S.a.W., Representation of Hypertext Documents Based on Terms, Links and Text Compressibility. 2010.

K. Suresh, D.K., S. Ghosh , S. Das and Ajith A., Automatic Clustering with Multi-Objective Differential Evolution Algorithms.

Knowles, J.H.a.J., Semi-Supervised feature selection via multiobjective optimization. 2006.

R. Neumayer, G.T.a.K.N., TRUMIT : A Tool to Support Large-Scale Mining of Text Association Rules. 2011.

Szymanski, J., Towards Automatic Classification of Wikipedia Content. 2010.

Szymanski, J., Self-Orgaanizing Map Representation for Clustering Wikipedia Search Results. 2011.

M. Mete, X. Xu, Chun-Yang F., Gal Shafirstein, Head and Neck Cancer Detection in Histopathological Slide, International Workshop on Data Mining in Bioinformatics, Sixth IEEE International Conference of Data Mining (ICDM 2006), December 18-22, 2006, Hong Kong.

Use of Chatterbot for Accessing Learning Objects on Mobile Devices With a Data Mining Search Engine.

Edgar Alan Calvillo Moreno[1], Miguel Meza [1]
Universidad Autónoma de Aguascalientes
Mexico[1]
alancalvillo@yahoo.com,
mmeza2000@hotmail.com

Jaime Muñoz Arteaga[1], Alejandro Padilla[1], Felipe Padilla[2],Francisco Javier Alvarez Rodriguez[1]
Universidad Autónoma de Aguascalientes
Mexico[1] , Ecole de Technologie Superieure
Canada[2]
jmauaa@gmail.com, apadilla@correo.uaa.mx,
fpadilla2000@hotmail.com,
fjalvar@correo.uaa.mx

***Abstract*—The use of learning objects across multiple platforms and the creation of learning repositories that are used to direct access into the learning objects and the deployment has been generated a new need in applying information, they are considered as educational resources that can be employed in technology-support learning, just like the use of a chatterbot, with his own search engine to locate learning objects using data mining.**

Keywords-Learning Object, Data Mining, Search Engine.

INTRODUCTION

The use of learning objects as resources are focused in the use of a virtual learning environment. The use of metadata with each learning object allows the possibility of use of all the information to create courses. The Learning Objects are typically stored in repositories that provides a list of services like view, download and updates.

This work includes the use of learning objects and the user has to work with digital media to learn , previously we need to use a chatterbot including basic information in a chat to get a correct resource as learning object. There are many tools for this kind of knowledge bases, but no one have been adapted to use as teaching resources including learning objects.

The University Autonomous of Aguascalientes(UAA) was working to create learning objects and it's member of the Latin American Community of Learning Objects; we proposed the use of the REDOUAA repository, it has hundreds of learning objects under SCORM standard. There are objects to make testes and create online courses in different areas. The REDOUAA allows display and storage learning objects and also offers online service to facilitate collaboration through users like forums, chats, wikis and collaborative editors.

Current work presents an important collection of information that may be included in content that will allow the chatterbot to access. Next section present sets out the possibility of a user that allows to access to a range of learning objects obtained via chatterbot, based on a basic pattern of search using the text introduced to generate a search in the repositories.

PROBLEM OUTLINE

The use of mobile devices to communicate and search information in the web generates the need of the exploitation of shared resources used by teachers in the UAA. That's resulted on the increased of resources available in other media like learning objects , this creation delis derived from the needs generated a considerable increase in the number of online courses. The constant work on the creation of resources led to the UAA to belong to a community of learning objects, establishing multiple connections between other learning object repositories, these resources must be used by the application, this function must focus on an intelligent environment that allows the exploitation of multiple learning objects.

The use of new techologies like media, html docs, pdf files, images, video in education generates multiple educational resources and with this complicate the correct selection of content to display for the user, it is necessary to have an application that can display information according to the user profile, that information can be obtained using a tool that

allows interaction between the user to obtain related patterns of information. We need these contents are usually created to be available via web browser and be connected to the Web, it is necessary to use these resources without an internet connection in particularly this works try to solve the following issues[11]:

- The creation of an architecture that enables the use of learning objects in a multiplatform device.
- The use of a chatterbot to create a conversation and have information to the searches.
- Create a research model to create an interconnection between multiple devices mobile and desktop computers.
- The performance of the application must be optimums in multiple searches.
- The generation of a multiple interface that works into mobile devices and desktop computers.
- The use of a specification to search information based on a metadata.
- The visualization of the learning objects.

CONTRIBUTION

The use of multiple communication devices, create a need to the use of shared resources apply into different kind of devices and operating systems, we propose in this document build an application that works in mobile devices and desktop computers.

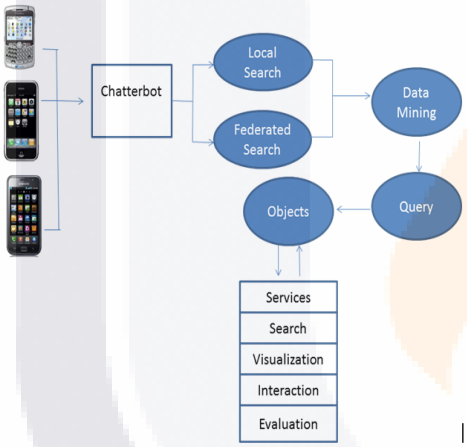

Fig. 1 Chat Interaction Model into a learning object repository.

The main architecture proposed based on the use of agents that allow the users interact with the chatterbot. The first step is the mobile device with a web access to the application and with a graphical interface adapted that allows you to send information to the chatterbot , after some questions, establish a standard process to ask the object list, consult the search services and finally obtaining the information required for optimal student learning

The interaction established between mobile devices and the chatterbot also you can check the type of search that is performed in case of a local or federated, and the adaptation of these results to the type of mobile device, taking into account that the information can come device operating system android, ios or blackberry . Once the type of device generates a list of learning objects based on the services offered by each object, such as lessons content, evaluation, interaction and those elements that increase the user learning.

We use the learning method contained into the learning object and the chatterbot as a tool of online education that allows the user to have constant feedback between the mobile device and platform that generate a list of learning objects that enable the user learn through practice and synthesized knowledge.

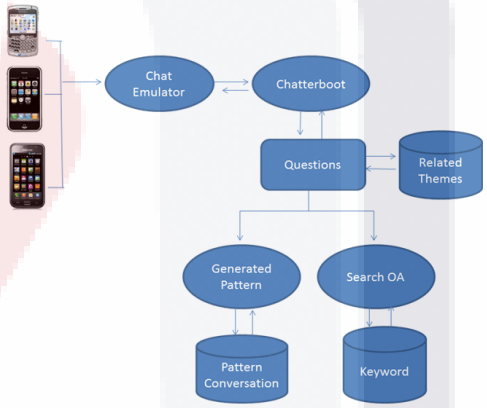

Fig. 2 The Internal Search Model in the Architecture

The Figure 2 shows the internal search using a chat to interact between mobile devices, this process does not include the device adapter and we assume you are working without any problems our chat emulator is the first in basic functions and a series of questions to assist the following process to work in this case the chatterbot use basic questions to get patterns and search into related themes that finally allows a query to our pattern conversation and the perform a search with keywords in our learning object repositories.

The chat interaction model (fig 1) works around a simulated chat environment that give us the ability to interact with our mobile device with a series of basic questions to determinate a particular area of consultation that will help us to generate keywords to create queries into our data base with data mining.

The internal search model(fig 2) will enable us to generate patterns of conversation that is stored on a database linking the interests of the first users to consult information in our repository of learning objects, this information will enable us to consult in the future with the constant of the application generating knowledge focused on patterns created by the users.

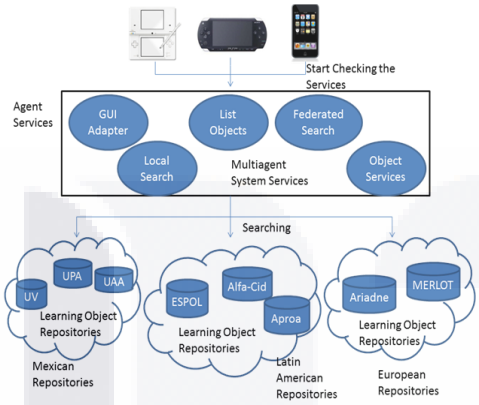

Fig. 3Federated Architecture of Learning Objects .

The Federal Architecture of Learning Objects [1] leads to the conclusion that we can define a federated search [5] as a web service as a result gives us a series of learning objects based on the search request by the user. These searches range are from a simple search within a web browser where you specify certain attributes to the search, based primarily on information contained in the metadata of learning objects stored, which will allow use in distance learning platforms reflected in various institutions belonging to the federation, all these services are performed independently of the instructions that are in the process of forming the federation.

CASE OF STUDY

The java application deployment on mobile devices has led to substantial increase in applications designed specifically for the deployment of different applications in this case we use an application designed for education, it use a repository of learning objects where there are resources specifically for faiths dimensions using a mobile device.

The prototype application was developed in Java to make easy the portability of the application and create the chatterbot that allows interaction between the user and the application, generating a first emulated chat where you can get basic questions about topics that allow the creation of a pattern in order to search and display of related information.

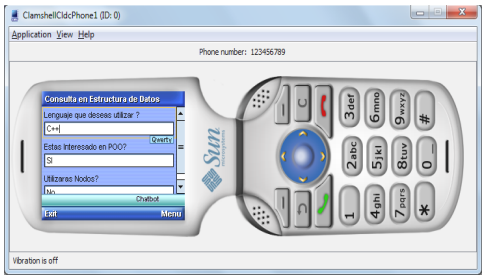

Fig. 4Chatterbot working on mobile devices

The chatterbot on mobile devices shows a first approach to the emulation of a conversation making some questions to define the issue in which you will work, in this case the structure in order to subsequently perform basic questions about the language used, the type of programming to use and a final question which asks if nodes where used for the development of practice with this series of questions, establishing a pattern of information that will help us to generate a list of learning objects that are displayed in the mobile device.

Fig. 5 The Learning Objects result.

The figure 5 shows a search result based on the information captured by our chatterbot with that obtained data needed to perform a query into the learning objects federation appling data mining and finally display into the device the result of the search.

<imsmd:general>

<imsmd:identifier>001 ED</imsmd:identifier>

<imsmd: weight >low</imsmd: weight >

<imsmd:title>

<imsmd:langstring xml:lang="en">Estructura de Datos</imsmd:langstring>

</imsmd:title>

<imsmd:catalogentry>

<imsmd:description>

<imsmd:langstring xml:lang="en”>Ejercicio sobre nodos en c++ </imsmd:langstring>

</imsmd:coverage>

Fig. 6 The Metadata with information used in searches and the field of weight used by the data mining algorithm.

This information is based on search and it consists of a metadata that contains each learning object, generating a series of fields for object with this structure we can use a database to help in the multiple search in the learning object repositories. That search makes using an algorithm of clusters that helps to increase the performance of the search using the field of weight stored firstly in the metadata of the learning object.

The application works usinge a cluster implementation and creates a division in the information based on the metadata field weight that helps to determinate the division of clusters based on the importance of the learning object.

RELATED WORK

There are a lot of works like the proposed in this article, but the chatterboot proposed are focus to get information from a simulated conversation and detect some patterns in the text introduced to make searches into the learning object repositories.

Works	***Intelligent Agent***	***Chatterbot***	***XML Reader***
[9]Grainne Conole	X		
[10]Ewerton Jose Wantroba		X	X
[11]Michelle Denise Leonhart	X	X	X
[12]Michell L. Mauldn		X	

Table. 1 Work Related.

The related work shows in the table 1 how all the presented works are focus to the use of a simulated chat but without any practical, we use the chatterbot to detect patterns in the conversation and make searches in the learning object repositories.

CONCLUSIONS

This work shows the beginning of an application that combines the use of learning objects primarily operated by a chatterbot using the information introduced by the user and searching some patterns of knowledge and with this information we can generate queries in the learning object repository implementing a cluster algorithm into the data base search to have a best time to get the list of learning objects.

The main goal of this article is to show an architecture that allows to exploit the benefits of an interface that allows constant communication between a student and the mobile device through a simulated chat get answers necessary to show the best possible results on screen.

It is necessary to develop an intelligent virtual learning environment using distributed learning object repositories from different institutions and it is necessary also portable devices to communicate between the students and the teacher.

FUTURE WORK

It is necessary an intelligent virtual learning environment using distributed learning object repositories from more institutions, actually there's only a few institutions added in the federated community and it is necessary also portable devices to communicate between the students and the teacher.

REFERENCES

E.A. Calvillo Moreno , F. Álvarez Rodríguez,J. Muñoz Arteaga,. and F. Martínez Ruiz, An Architectural Model in base in Mobile Devices for the Latin American Federation of Objects of Learning, LACLO 2008 (3ra. he/she confers Latin American of Objects of Learning), October 28. 31 2008, Aguascalientes, Mexico, ISBN 978-970-728-067-0

Jacsó, Péter, Thoughts About Federated Searching, Information Today, Oct 2004, Vol. 21, Issue 9.

S. Heras, M. Rebollo, V. Julián, Arguing About Recommendations in Social Networks, IAT 2008: 314-317.

C. Lopez Guzman, F. Garcia Peñalevo, Formation of learning Objects through the use of the metadatos of a digital collection: of Dublin Core to IMS,SPDECE 2004, October 20. 24 2008, Guadalajara, España.

A Ochoa Zezzatti ,A. Padilla, González, S. Castro, A. & Shikari Hal : Improve a Game Board Base on Cultural Algorithms. IJVR 7(2): 41-46 (2008)..

J. Muñoz Arteaga, X. Ochoa, E. A. Calvillo Moreno and Gonzalo Vine, Integration of REDOUAA to the Latin American Federation of Repositories of Learning Objects, LACLO 2007 (2da. he/she confers Latin American of Objects of Learning), October 22. 25 2007, Santiago from Chil.

M. Nikraz, G. Caire and Parisa TO. Bahri, TO Methodology for the Analysis and Design of Multi-Agent System Using Jade.

Xavier Ochoa, ESPOL, http://www.ariadne-eu.org/index.php., October 2009 as accessed date.

Grainne Conole, The Role of Mediating Artefacts in Learning design,IGI Global Distributing 2008.

Ewerton Jose Wantroba, Um exemplo do uso padrao XML na definicion de um linguage espezializada para IA.

Michelle Denise Leonhart, ELEKTRA : Um Chatterbot para Uso em Ambiente Educacional

Michell L. Mauldn, Chatterbots Tiny Muds and the Turing Test Entering the Loebner Prize Competition, Carnegie Mellon.

Pasquale De Meo, Alfredo Garro, Giorgio Terracina, Domenico Ursino : X-Learn : An XML-Based, Multi-agent System for Supporting "User-Device" Adapative E-Learning. CoopIS/DOA/ODABASE 2003:739-756

Robert Costello, Darren P. Mundy : The Adaptive Intelligent Personalised Learning Environment . ICALT 2009:606:610