SISTEMA DE INDEXACIÓN DE DOCUMENTOS DE JURISPRUDENCIA BASADO EN INTELIGENCIA ARTIFICIAL

EDIER ANCHICO SILVA

Trabajo de grado

Ingeniería de Sistemas

FUNDACION UNIVERSITARIA DE POPAYÁN
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
GRUPO DE INVESTIGACIÓN IMS

Popayán, abril de 2019

SISTEMA DE INDEXACIÓN DE DOCUMENTOS DE JURISPRUDENCIA BASADO EN INTELIGENCIA ARTIFICIAL

EDIER ANCHICO SILVA

Trabajo de grado

Ingeniería de Sistemas

Director: Cristian Camilo Ordoñez Quintero

Codirector: José Armando Ordóñez

FUNDACION UNIVERSITARIA DE POPAYÁN

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

GRUPO DE INVESTIGACIÓN IMS

Popayán, abril de 2019

DEDICATORIA

A mi madre por ese apoyo incondicional y el esfuerzo que hizo por sacarme adelante. A mi familia por el tiempo que dejamos de compartir con ellos mientras cumplíamos con nuestro deber.

AGRADECIMIENTOS

Al Ing. Cristian Camilo Ordoñez Quintero y Adriana Eugenia Muñoz por su gran apoyo.

TABLA DE CONTENIDO

LISTA DE ILUSTRACIONES

LISTA DE TABLAS

RESUMEN

En Colombia el precedente judicial o Jurisprudencia facilita la toma de decisiones por parte de los jueces basándose en sentencias anteriores para poder dar un veredicto. Es por ello que los profesionales del derecho deben buscar entre un gran número de documentos las sentencias que ayuden como soporte para sus casos en ejecución donde se soportan en diferentes motores de búsqueda y aplicaciones para tener los documentos que puedan solventar las necesidades del caso. En la actualidad el uso de inteligencia artificial y métodos de procesos de lenguaje natural hace que los diferentes sistemas sean más óptimos a la hora de dar solución a un problema dado, por ello dentro de este trabajo se propone y evalúa un sistema de búsqueda de documentos judiciales soportado en inteligencia artificial, todo para optimizar los procesos de búsqueda y análisis de dichos documentos judiciales, donde se obtiene como resultado que la precisión de los métodos de búsqueda es prometedora y satisfactoria para los usuarios.

Palabras clave: Jurisprudencia, inteligencia artificial, Gensim, Python, PLN (Procesamiento del Lenguaje Natural), NLTK.

ABSTRACT

In Colombia, the judicial precedent or jurisprudence facilitates the decision making by the judges based on previous judgments to be able to give a verdict. That is why law professionals must search among a large number of documents for judgments that help as support for their cases in execution where they are supported in different search engines and applications to have the documents that can solve the needs of the case. Currently the use of artificial intelligence and natural language process methods makes the different systems more optimal when it comes to solving a given problem, therefore within this work is proposed and evaluated a document search system judicial processes supported in artificial intelligence, all to optimize the search and analysis processes of said judicial documents, where the result is that the accuracy of the search methods is promising and satisfactory for the users.

Keys words: Jurisprudence, artificial intelligence, Gensim, Python, PLN (Natural Language Processing), NLTK.

INTRODUCCIÓN

Las leyes 1437 y 1564 referentes al código contencioso administrativo y de procedimiento administrativo y código general del proceso, reconocen la importancia del precedente judicial para la seguridad jurídica de los administrativos y exigen su aplicación [1], mediante la aplicación de precedentes judiciales que hubieran resuelto casos similares al suyo teniendo en cuenta el artículo 10 del código contencioso administrativo [2]. Por su parte el artículo 102 de la norma ya citada establece la extensión de la jurisprudencia a terceros, tal que las autoridades públicas deben extender los efectos de una sentencia de unificación jurisprudencial a quienes lo soliciten y acrediten los mismos supuestos facticos y jurídicos [3].
En los conflictos sometidos al conocimiento de una autoridad jurisdiccional o administrativa, en los que se pretenda acudir al precedente, es necesario identificar decisiones judiciales anteriores o precedentes que se observan sobre hechos similares de los que se estén resolviendo actualmente, así como debe distinguirse con claridad en cada una de ellas y los argumentos jurídicos que fundamentaron la decisión que puedan aplicarse en el caso en cuestión. Es evidente que determinar el precedente judicial para un caso concreto, es una ardua tarea que implica estudio a profundidad de gran cantidad de textos jurisprudenciales, tanto para los funcionarios judiciales y administrativos en la resolución de conflictos, como para los abogados y usuarios de la administración de justicia que constantemente buscan argumentos en pro de sus intereses; es así como se reconoce la necesidad procesar este cúmulo de información de una forma sencilla y ágil; que aunque ya cuentan con algunas herramientas puestas a su disposición, estas no ofrecen una solución real al problema debido a que su funcionamiento se limita a operaciones básicas de búsqueda.

Actualmente, se han desarrollado algunas herramientas tecnológicas cuya efectividad es aún baja [4], y existen en curso igualmente, algunas aproximaciones orientadas al uso del procesamiento de lenguaje natural en ámbitos legales [5]. En este escenario, las tecnologías de recuperación automática de información

representan una buena aproximación inicial a la solución del problema. La recuperación de información está compuesta por tareas de búsqueda y consulta de información, clasificación de resultados, optimización de la representación y almacenamiento de la información, clasificación de documentos en grupos pre definidos y agrupamiento de documentos en conjuntos definidos a partir del análisis automático del contenido del documento (clustering) [6], Sin embargo estas aproximaciones no han abordado el dominio específico del precedente judicial en Colombia.

CAPÍTULO 1

1. ASPECTOS GENERALES DE LA INVESTIGACIÓN

1.1 PLANTEAMIENTO DEL PROBLEMA

Para facilitar los procesos de búsqueda e identificación de información, es común encontrar herramientas basadas en procesamiento de lenguaje natural (NLP, por sus siglas del inglés) la cual es la habilitad que puede tener una máquina para procesar información comunicada y entender su significado [7]. El procesamiento del lenguaje no sólo engloba las complejidades inherentes del procesamiento del lenguaje natural ordinario; sino también, tiene en cuenta las características específicas del dominio, haciendo del procesamiento de documentos particulares de un dominio un desafío que demanda soluciones específicas [7]. En Colombia y en muchos otros países, la capacidad que tiene el juez para atemperar sus decisiones y la interpretación de la norma a las circunstancias propias del caso, ocupa en este momento un lugar fundamental en la función judicial, por lo tanto, un juez no sólo se limita a la mera aplicación de normas vigentes, sino que, por el contrario, acude a justificaciones de su propio razonamiento. Así las cosas, el derecho observa necesario imponer criterios, límites y técnicas que garanticen la seguridad jurídica y la igualdad en el acceso a la administración de justicia, para que los individuos no estemos indefensos frente a la subjetividad de los jueces, siendo una de estas técnicas el precedente judicial [8]. En los conflictos sometidos al conocimiento de una autoridad jurisdiccional o administrativa, en los que se pretenda acudir al precedente, es necesario identificar decisiones judiciales anteriores o precedentes que versen sobre hechos similares a los que se estén resolviendo actualmente, así como debe distinguirse con claridad en cada una de ellas, los argumentos jurídicos que fundamentaron la decisión que puedan aplicarse en el caso en cuestión. Es necesario tener en cuenta que las decisiones judiciales tienen un efecto diferente dependiendo de la autoridad que las profiera, por lo cual

dichos fallos habrán de discriminarse por jurisdicción, especialidad y jerarquía del fallador.

Es evidente que determinar el precedente judicial para un caso concreto, es una ardua tarea que implica estudio a profundidad de gran cantidad de textos jurisprudenciales, tanto para los funcionarios judiciales y administrativos en la resolución de conflictos, como para los abogados y usuarios de la administración de justicia que constantemente buscan argumentos en pro de sus intereses; es así como se reconoce la necesidad procesar este cúmulo de información de una forma sencilla y ágil; que aunque ya cuentan con algunas herramientas puestas a su disposición, estas no ofrecen una solución real al problema debido a que su funcionamiento se limita a operaciones básicas de búsqueda, son lentas y además ninguno de ellos realiza una búsqueda exacta, debido a lo anterior se piensan realizar pruebas con diferentes herramientas capaces de agilizar estos procesos basados en la fluidez y la precisión, comparando elementos clave de cada sentencia sin olvidar su contexto las cuales servirán para categorizar cada búsqueda, de procesos de obtención de sentencias dadas con anterioridad ahorrando tiempo para los funcionarios como para los ciudadanos. Algunos mecanismos de inteligencia artificial han sido usados en diversos campos para el análisis de la información, particularmente las áreas de procesamiento de lenguaje natural y la minería de texto. Sin embargo, estas aproximaciones no han abordado el dominio específico del precedente judicial en Colombia; la ventaja de realizar pruebas con diferentes algoritmos de inteligencia artificial es que, al realizar un análisis minucioso de cada una, se logrará elegir la más adecuada para solucionar la problemática mencionada con el fin de que la búsqueda sea precisa y tenga un alto rendimiento.

1.2 JUSTIFICACIÓN

Frente a las necesidades del derecho, es común acudir a la denominada informática jurídica, definida por Julio Téllez Valdés [9] "como la técnica interdisciplinaria que tienen por propósito la aplicación de la informática a la recuperación de información jurídica, así como la elaboración y aprovechamiento de instrumentos de análisis y tratamiento de dicha información, necesarios para una toma de decisión con repercusiones jurídicas", por eso se desea desarrollar un buscador capaz de recuperar datos de forma eficaz y con un porcentaje de exactitud bastante elevado debido que en Colombia no existe una herramienta con éstas características que permita un uso eficiente en la recuperación de sentencias dadas por la jurisprudencia Colombiana. Debido a lo anterior es importante disponer de mecanismos que justificante para el desarrollo del proyecto es la alta prioridad que ha dado el estado colombiano a este tema, que se evidencia en la convocatoria estratégica de innovación para la justicia de parte del ministerio colombiano de las telecomunicaciones y Colciencias [1], en donde se resalta la importancia del precedente judicial para la seguridad jurídica de los administrativos y de los ciudadanos plantea las siguientes líneas temáticas:

➢ Generar las herramientas y aplicaciones que permitan la identificación y divulgación de Referentes de Unificación Jurisprudencial y Líneas jurisprudenciales.

➢ Generar las herramientas y aplicaciones que permitan la identificación y divulgación de las Ratio Decidenci "razón para decidir", estas son las razones de fondo que determinaron la solución dada al caso concreto de las sentencias de tutela y de constitucionalidad en las sentencias proferidas por la Corte Constitucional.

➢ Generar las herramientas y aplicaciones que permitan la identificación y divulgación de doctrinas probables, esto es, tres decisiones de la Corte Suprema como Tribunal de Casación sobre un mismo punto de derecho, según el artículo 7 del Código General del Proceso y el artículo 40 de la ley 153 de 1887.

Utilizar nuevas herramientas que mejore la velocidad de las consultas en un sistema es de vital importancia debido a que la tecnología está en constante evolución. El algoritmo de inteligencia artificial en su época no se podía ejecutar, hoy en día está funcionando y ha permitido que este campo de la IA (Inteligencia Artificial) se aproveche al máximo, gracias a ello se ha podido predecir enfermedades, conductas humanas, reconocimiento de imágenes entre otras.

Las consultas de sistemas mencionadas anteriormente, se limitan a buscar en una base de datos el texto que contenga esa palabra específica, en cambio la inteligencia artificial aprende de estas consultas a medida que el usuario las realiza ofreciendo una consulta más rápida y eficiente, dejando atrás aquellos sistemas que se limitaban a los mismos resultados de búsqueda.

Por lo tanto, se genera la idea de implementar un sistema de indexación de documentos de jurisprudencia basado en inteligencia artificial, el cual facilite la búsqueda y recuperación de documentos jurisprudenciales.

Es muy importante resaltar que este documento es útil para los abogados e investigadores, ya que sirve para proporcionar más conocimiento conceptual y practico con relación a la búsqueda de indexación para el procesamiento de documentos jurisprudenciales, para las diferentes organizaciones en general.

1.3 APORTES DE LA INVESTIGACION

- Registro de software: Como parte del proyecto se realizó el registro de software además de secreto empresarial con la empresa vinculada a este proyecto donde se protege el sistema desarrollado para evitar plagio.
- Monografía: Es el trabajo realizado donde se almacena toda la información relacionada con el desarrollo del sistema.

- Artículo: Artículo de investigación enviado a revista donde se anexa todo el desarrollo científico de esta propuesta.
- Aplicación: En esta se encuentra la aplicación realizada para poder realizar búsqueda de documentos de jurisprudencia. http://34.205.27.198/

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Desarrollar un sistema basado en inteligencia artificial que facilite la búsqueda y recuperación de documentos jurisprudenciales

1.4.2 OBJETIVOS ESPECÍFICOS

- Definir mecanismos de búsqueda e indexación para el procesamiento de documentos jurisprudenciales.
- Crear un repositorio de documentos de prueba para entrenar los algoritmos de inteligencia artificial.
- Desarrollar y evaluar el sistema de indexación de documentos jurisprudenciales, por parte de expertos.

1.5 ORGANIZACIÓN DEL DOCUMENTO

CAPITULO1 - ASPECTOS GENERALES: Se presentan aspectos generales como el planteamiento del problema, justificación, aportes que realizará el proyecto, objetivos generales y específicos que se cumplen durante la ejecución del mismo.

CAPITULO2 - MARCO DE REFERENCIA: Se presenta el marco conceptual el cual describe los conceptos más relevantes del proyecto y el estado del arte que contiene los resultados de otras investigaciones similares al tema de investigación del proyecto actual.

CAPITULO3 - CARACTERÍSTICAS DE ALGORITMOS DE PROCESAMIENTO DE LENGUAJE NATURAL: Se presentan las características de los algoritmos implementados para el sistema de recuperación de documentos jurisprudenciales, los métodos utilizados y las etiquetas.

CAPITULO 4 - REPOSITORIO DE DOCUMENTOS DE PRUEBA PARA ENTRENAR LOS ALGORITMOS DE INTELIGENCIA ARTIFICIAL: Se presenta el proceso de implementación de mecanismos para la creación de la base de datos para realizar la búsqueda de documentos de jurisprudencia

CAPITULO 5 – VALIDACIÓN DEL SISTEMA POR EXPERTOS: Se presenta el proceso de evaluación y pruebas realizadas por expertos en el ámbito judicial de tal manera que la búsqueda logre cumplir los objetivos propuestos.

CAPITULO 6 – BIBLIOGRAFÍA: Por último, se presenta la bibliografía la cual contiene el conjunto de referencias utilizadas para el proyecto.

CAPÍTULO 2

2. CONTEXTO TEORICO Y ESTADO DEL ARTE

2.1 MARCO CONCEPTUAL

La jurisprudencia son los derechos por los cuales los jueces pueden tomar decisiones en base a fallos pasados para dar una sentencia actual según el precedente y cuando el caso tenga similitud con los hechos anteriores [10], esto se traduce en mayor agilidad sobre el proceso y una línea general al momento de tomar decisiones en la sentencia. Este documento tratará la implementación de una herramienta tecnológica para poder agilizar y dar mejores resultados al momento de buscar estas sentencias previas y poder encontrar la más adecuada. ISR (Information Search and Retrieval) o en español Búsqueda y Recuperación de Información [11], la cual es la parte de la informática que se encarga de la búsqueda de información en documentos electrónicos o cualquier medio digital, video, imágenes, audios y su objetivo es la recuperación de este material y mostrar su información de forma escrita y muy relevante, este proceso se ve reflejado en diferentes herramientas de uso cotidiano como son los buscadores[8] que normalmente usamos por ello dentro de este documento definimos una serie de conceptos relevantes a tener en cuenta en esta propuesta.

2.1.1 LSA (LATENT SEMANTIC ANALYSIS)

El análisis semántico latente planteado por Deerwester y colegas en 1988, es un método en el procesamiento del lenguaje natural, su función es analizar las relaciones entre un grupo de palabras que contiene al producir un grupo de conceptos relacionados con otros diferentes contextos. El método emplea disminuir la dimensionalidad con una técnica llamada descomposición de valores singulares(SVD) (Deerwester y colegas en 1990), esta se utiliza para detectar un espacio semántico latente, donde las palabras del mismo contexto semántico suelen aparecer juntas o en similares contextos del dominio. Este modelo semántico

emerge con la finalidad de vencer los problemas de la semántica, que son producidas por la sinonimia y la polisemia de las palabras en el contexto aplicado[12].

El algoritmo LSA es muy utilizado en las siguientes aplicaciones:

- Comparación de los documentos en el espacio de baja dimensión (agrupación de datos, clasificación de documentos).
- Encontrar documentos similares en todos los idiomas, después de analizar un conjunto básico de documentos traducidos (recuperación en varios idiomas).
- Encontrar relaciones entre términos (sinonimia y polisemia).
- Dada una consulta de términos, tradúzcala en el espacio de baja dimensión y encuentre los documentos correspondientes (recuperación de información).
- LSA se ha utilizado para ayudar en la realización de búsquedas en la técnica anterior para patentes.

2.1.2 JURISPRUDENCIA

La jurisprudencia es la encargada de establecer la doctrina que desarrolla el tribunal en los distintos ámbitos del derecho, esto es, cada caso en concreto que resuelve la autoridad, en el que se evidencian diversos temas. Según el criterio de la doctrina, se puede establecer que la jurisprudencia es un criterio auxiliar, en el sentido que, cuando las disposiciones de la Constitución y de las demás fuentes formales del derecho no tienen un sentido unívoco, que sea capaz de eliminar toda indeterminación, la jurisprudencia auxilia al entendimiento pleno del sentido de dichas fuentes formales, pues en ella se encuentran las normas adscritas que expresan su significado en sentido prescriptivo [10].

2.1.3 BÚSQUEDA Y RECUPERACIÓN DE INFORMACIÓN (ISR)

Es un proceso articulado y, en muchas ocasiones, retroalimentado que se inicia cuando una persona tiene un problema que quiere resolver mediante la obtención de cierta información que termina cuando la persona resuelve este problema con la información obtenida y que se implementa a través de la identificación y localización de los documentos que contienen esta información que es pertinente para satisfacer la necesidad de información de la persona [11].

2.1.4 BUSCADOR

Un buscador es un sistema informático que nos permite encontrar páginas web o resultados en base a la frase o palabra que se haya ingresado y se esté buscando. La cadena de texto ingresada por el usuario va desde lo más general a lo más específico, el objetivo es que mediante un algoritmo encuentre la solución a lo que el usuario busca. [13].

2.1.5 RATIO DECIDENDI "razón para decidir"

Comprende el alcance de las disposiciones jurídicas, exponiendo qué es lo que se prohíbe, permite, ordena o habilita para el caso específico [10]**.**

2.2 ESTADO DEL ARTE

En Colombia, las Altas Cortes que dirigen las tres jurisdicciones principales, a saber, Constitucional, Ordinaria y Contencioso Administrativa, tienen dispuesto herramientas de búsqueda jurisprudencial que brindan al usuario opciones de búsqueda tomando en cuenta elementos como número de radicado, año, magistrado, entre otros, destacándose la posibilidad de buscar jurisprudencias por texto o palabras clave.

La corte suprema de justicia tiene a disposición de todos los interesados 2 herramientas para la búsqueda de sentencias. En primer lugar, se cuenta con la herramienta denominada “Consulta Jurisprudencial” este da la posibilidad de filtrar búsquedas por Salas según la especialidad, año, magistrado ponente (quien redactó la sentencia), por número de radicado de la Corte o por texto libre dentro de las providencias. En esta última opción se permite que el usuario introduzca un texto que contenga el tema requerido para la detección de sentencias útiles para su caso, la herramienta buscará entre todas las sentencias de la Corte aplicando los filtros que se mencionaron anteriormente y presentara al usuario una lista que contiene todas las sentencias acordes con los parámetros de búsqueda ingresados por el usuario, además da la posibilidad de leer un resumen o descargar el texto completo del documento; Un aspecto importante a señalar, es que el resumen ofrecido por el sistema presenta deficiencias, pues se limita a presentar oraciones en donde se resaltan los parámetros de búsqueda inicialmente ingresados por el usuario, que en la mayoría de los casos no pueden ser leídos con coherencia o continuidad.

Por otra parte, la corte suprema, ofrece 3 índices temáticos sobre Habeas Corpus, Sistema Acusatorio y Ley de Justicia y Paz, cada índice consta de una tabla en donde se muestra el nombre, año y número de radicado de cada una de las sentencias, que a su vez se encuentran clasificadas en una serie de subtópicos. La desventaja más evidente radica en que no cuenta con un sistema de búsqueda, obligando al usuario a buscar manualmente dentro de cada subtópico.

El Consejo de Estado, máximo tribunal de la jurisdicción contencioso administrativa, ofrece una herramienta conocida como Consulta de Jurisprudencia, permite realizar

búsqueda de jurisprudencia tomando en cuenta factores como los ofrecidos por la Corte Suprema de Justicia, pero agrega varios como Sección, Tipo de Providencia, Fecha, Partes, etc. Un punto a diferenciar es que adicionalmente cuenta con un filtro por “tema”, que corresponde al resumen general de la sentencia, sin embargo, al revisar los resultados se observa que este resumen aparece en blanco, por lo cual el usuario deberá descargar el texto completo de la sentencia para enterarse sobre su contenido y poder decidir si le es útil para sus necesidades.

La corte constitucional, máximo tribunal de la Jurisdicción Constitucional, pone a disposición de los usuarios una herramienta web, entrega al usuario varias opciones de búsqueda: i) por Número de Providencia, en el caso que se pretenda descargar una sentencia en particular ya identificada; ii) por radicado, cuando se pretende descargar una determinada sentencia pero se desconoce el número, para lo cual se solicitan datos como magistrado ponente, tema, año, sentencia y demandado; iii) por índice temático, en el cual se puede introducir una o varias palabras que se identifiquen con un índice elaborado por la propia corte, desplegándose la lista de sentencias con los temas seleccionados que ofrece la Corte; iv) finalmente por texto libre dentro de las providencias, que le permite al interesado escribir un texto que se refiera a un tema que las sentencias requeridas deban tratar , además da la posibilidad de aplicar filtros por año. En esta última opción, la herramienta buscará entre todas las sentencias de la Corte dentro del año señalado, después de lo cual desplegará una lista que contiene sentencias que dentro de sus cuerpos contengan palabras clave relacionadas con el texto solicitado, ofreciendo por cada sentencia la posibilidad de leer un resumen o descargar el contenido total de la sentencia. Se observa que, si bien la búsqueda se hace por el texto introducido, en algunos casos aparecen sentencias que no responden a la totalidad de los criterios seleccionados, pero que sí contienen parte del texto solicitado por el usuario, lo que genera que aparezcan sentencias que aun cuando están relacionadas con el tema no responden estrictamente a lo que el interesado necesita.

Una de las características que comparten las herramientas nombradas, es su funcionamiento básico en las búsquedas; incapaces dar respuestas precisas a los

requerimientos de los usuarios, falencia que puede ser superada con la aplicación de sistemas de inteligencia artificial como el procesamiento de lenguaje natural.

El principal objetivo del procesamiento de lenguaje natural, es mejorar la interacción de las personas con los sistemas informáticos, que con el uso de tecnologías de inteligencia artificial son capaces de entender el lenguaje que usan las personas para comunicarse. El procesamiento de lenguaje natural hace posible y ha sido usado para desarrollar sistemas capaces de procesar cierta cantidad de información (en forma de texto) y determinar de manera muy precisa las palabras clave, generar resúmenes, traducciones, etc. Algunas de las herramientas disponibles para el desarrollo de aplicaciones basadas en procesamiento de lenguaje natural son:

GATE (General Architecture for Text Engineering) (The University of Sheffield, n.d.): herramienta de código abierto para la anotación semántica que además cuenta con una serie de componentes compatibles con diferentes idiomas, que incluye reconocimiento de entidades (NER, por sus siglas del inglés Name Entity Recognition), extracción de información, herramientas para el manejo de ontologías, anotación semántica, etc. Los recursos ofrecidos por GATE están clasificados en 3 categorías, i) Recursos de lenguaje: que agrupan los módulos para el análisis superficial del lenguaje natural; ii) Recursos de procesamiento: representados por algoritmos; iii) Recursos visuales: en donde se agrupan los componentes de la interfaz de usuario.

Stanford CoreNLP [14] provee una serie de herramientas de código abierto escritas en java, para el análisis de lenguaje natural; es una herramienta que posee un alto grado de flexibilidad y extensibilidad que integra herramientas para POS tagging, NER, parsers, etc. Originalmente desarrollado para el procesamiento de recursos en ingles pero que en sus últimas versiones ofrece varios niveles de soporte para el español, chino, francés, alemán y árabe como también.

OpenNLP [15] Herramienta que utiliza algoritmos de aprendizaje automático para el procesamiento de lenguaje natural, que está en capacidad de soportar las tareas comunes que conforman el NLP: Tokenizacion, segmentación de oraciones, POS tagging, NER, etc; tareas que usualmente son requeridas para la construcción de

servicios de procesamiento más avanzados, por otra parte Fersini y Sartori en [16] pretenden optimizar las búsquedas de ficheros multimedia resultado de audiencias judiciales, para lo cual hacen uso de las transcripciones realizadas en el transcurso del proceso legal, anotaciones sobre el audio / video realizadas por los usuarios y reconocimiento automático del habla; expansión de los términos de búsqueda ingresados por los usuarios, añadiendo nuevos términos relacionados al hacer uso de los conceptos vinculados en una ontología legal y agrupando los resultados de búsqueda, aplicando el algoritmo k-means por bisectriz inducida.

En [17], se plantea el uso de algoritmos de aprendizaje para mejorar los procesos de búsqueda de casos legales en portugués, apoyados en procesos de agrupación y categorización. Los autores del artículo definen un nuevo paradigma denominado “bolsa de términos y referencias legales”, usando un diccionario de términos del dominio legal para la detección de términos legales y el análisis de expresiones regulares para descubrir referencias legales, que en conjunto corresponden a las características principales que serán usadas como parámetros de agrupación. La formación de grupos se realiza por un proceso de división, en el cual inicialmente se define un grupo general y de manera iterativa se analizan los documentos contenidos en este, si la similitud del documento es baja con respecto al centroide del grupo, el documento es separado para formar un nuevo grupo, entre otras. El código de las herramientas de Procesamiento de Lenguaje Natural desarrolladas en Java y C++ es compilado a un código máquina de bajo nivel, lo que le confiere una mayor rapidez de computo. Esta amplia capacidad de procesamiento en comparación a herramientas para Python o Ruby es aprovechable ya sea en aumentar la cantidad de datos a procesar o reducir el tiempo que lleva el procesamiento de los mismos. Sin embargo, el que una herramienta este diseñada para un lenguaje de estas características tienen como consecuencia la reducción de flexibilidad inmediata de la herramienta, ya que el desarrollo de software en los lenguajes fuertemente tipados conlleva una mayor cantidad de tiempo a la hora de especificar la relación, jerarquía y tipado de los datos, con el fin de aprovechar el aumento en velocidad que ofrecen. Es por esta razón que se optó por lenguajes de guiones (scripting), como Python o Ruby, que, sin perder generalidad en cuanto a

la resolución de problemas, y a pesar de tener una velocidad de computo más reducida, permiten el desarrollo de soluciones adaptadas a los problemas que surgen con una mayor flexibilidad y rapidez, ofreciendo además una mayor cantidad de funcionalidad estándar para el procesamiento de texto [12]. Gensim es una biblioteca de código abierto y libre implementada en Python, que posee implementaciones de los algoritmos no supervisados como LSA y Random Projection para descubrir estructuras semánticas en documentos textuales y detecta tópicos con LDA. Presenta varios esquemas de peso como TF-IDF y posee compatibilidad con las bibliotecas de NumPy y SciPy. Permite usar el paradigma de computación distribuida para LSA y LDA y así acelerar los cálculos [18].

CAPÍTULO 3

3. DEFINCION DE METODOS DE BUSQUEDA DE DOCUMENTOS

3.1 ALGORITMOS

3.1.1 TFI-IDF

Es un algoritmo que cuenta el peso de la palabra considerando la frecuencia de la palabra (TF) y en cuántos archivos se puede encontrar la palabra (IDF). Como la IDF puede ver en cuántos archivos se puede encontrar un término, puede controlar el peso de cada palabra. Cuando una palabra se puede encontrar en tantos archivos, se considerará como una palabra sin importancia. Se ha demostrado que TF-IDF crea un clasificador que podría clasificar los artículos por ejemplo en las noticias en Bahasa Indonesia con una alta precisión; 98,3%[19].

3.1.2 LSA (LATENT SEMANTIC ANALYSIS)

El análisis semántico latente (LSA) es un método para analizar un fragmento de texto con cierta computación matemática y analiza la relación entre los términos en los documentos, entre los documentos en el corpus. Varias aplicaciones de recuperación de información inteligente, motores de búsqueda, sitios de noticias en Internet requieren una Método preciso para acceder a la similitud de documentos con el fin de llevar a cabo tareas de clasificación, agrupación, resumen o búsqueda. En este artículo, estudiaron el análisis semántico latente basado en la descomposición de un solo valor. El objetivo del análisis semántico latente es explotar la estructura global de los documentos. El énfasis es encontrar una relación oculta en el documento para comprender mejor el enlace entre los términos y el documento en el conjunto de datos. En este documento se realizó un estudio que utiliza el análisis semántico latente (LSA) para encontrar la correlación de términos en un conjunto de datos que consiste en trabajos de investigación de varias aplicaciones de procesamiento de lenguaje natural. La LSA muestra que la

descomposición de un solo valor colapsa varios términos con la misma semántica y puede identificar Términos con significado múltiple y representación de documentos en espacio conceptual tridimensional inferior[20].

LSA utiliza la descomposición de valores singulares (SVD) es un método de factorización que crea matrices de otras matrices de la original para su posterior procesamiento. Una matriz rectangular se descompone en el producto de otras tres matrices. Una matriz de componentes describe las entidades de fila originales como vectores de valores de factores ortogonales derivados, otro describe las entidades de columna originales en el mismo y la tercera es una matriz diagonal que contiene valores de escala de manera que cuando los tres componentes se multiplican por matriz, la matriz original se reconstruye. Hay una prueba matemática donde cualquier matriz puede descomponerse perfectamente, sin usar más factores que la dimensión más pequeña de la matriz original. Cuando menos que el número necesario De los factores que se utilizan, la matriz reconstruida es el mejor ajuste de mínimos cuadrados. Uno puede reducir el dimensionalidad de la solución simplemente eliminando los coeficientes en la matriz diagonal, Generalmente empezando por los más pequeños. (En la práctica, por razones computacionales, por muy grandes.
Los cuerpos sólo pueden ser un número limitado de dimensiones (actualmente unos pocos miles) construido [20].

$$\boldsymbol{M = U\,D\,V^{*}}$$

$\boldsymbol{M}$ es una matriz $M * M$

$\boldsymbol{U}$ es un matriz $M * N$ izquierda matriz singular

$\boldsymbol{D}$ es una matriz diagonal $N * N$ con números reales no negativos.

$\boldsymbol{V}$ es una $M * N$ derecha, matriz singular

$\boldsymbol{V^{*}}$ es $N * M$ matriz, que es la transposición de la V.

Matriz de identidad: es una matriz cuadrada en la que todos los elementos de la diagonal principal son unos y todos los demás elementos son ceros.

Matriz diagonal: es una matriz en la que todas las entradas que no son la diagonal principal son todas cero.

Matriz singular: una matriz es singular si su determinante es 0 o una matriz cuadrada que no tiene una matriz inversa.

A continuación, se observa el ejemplo para la utilización de LSA.

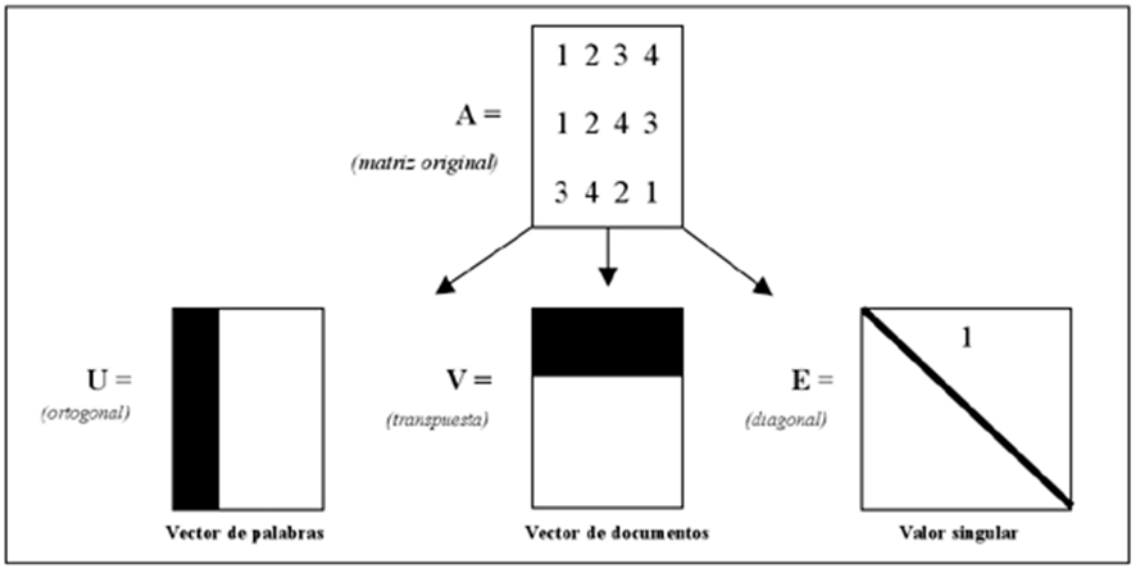

Ilustración 1 Representación de SVD tomado [21]

Descomposición de valores es que la matriz original se separa en 3 matrices.

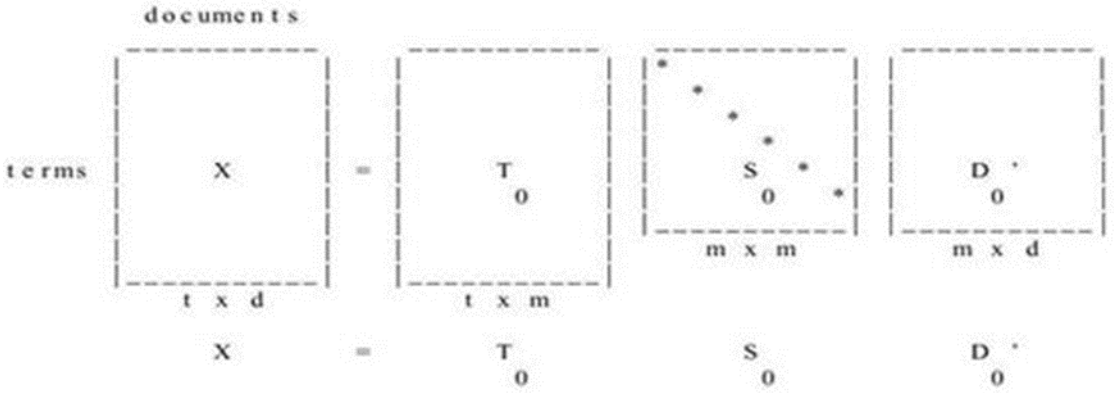

Ilustración 2 Matrices SVD tomado [22]

En esta ilustración se reduce las dimensiones a partir de la variable k que crea una nueva matriz con el número de k propuesto, que en todos los artículos citados varía entre 100 a 300 temas.

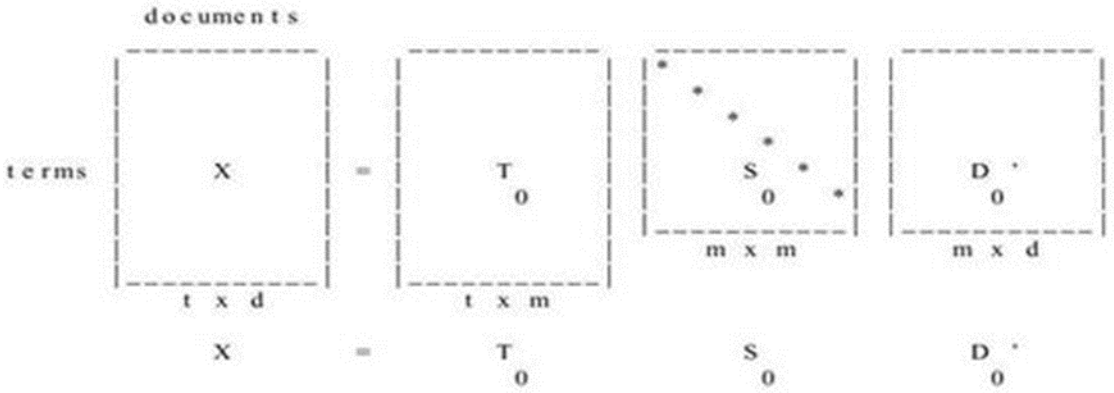

Ilustración 3 Matrices SVD K(temas) tomado [22]

En esta siguiente ilustración se muestra un ejemplo si k fuera 2.

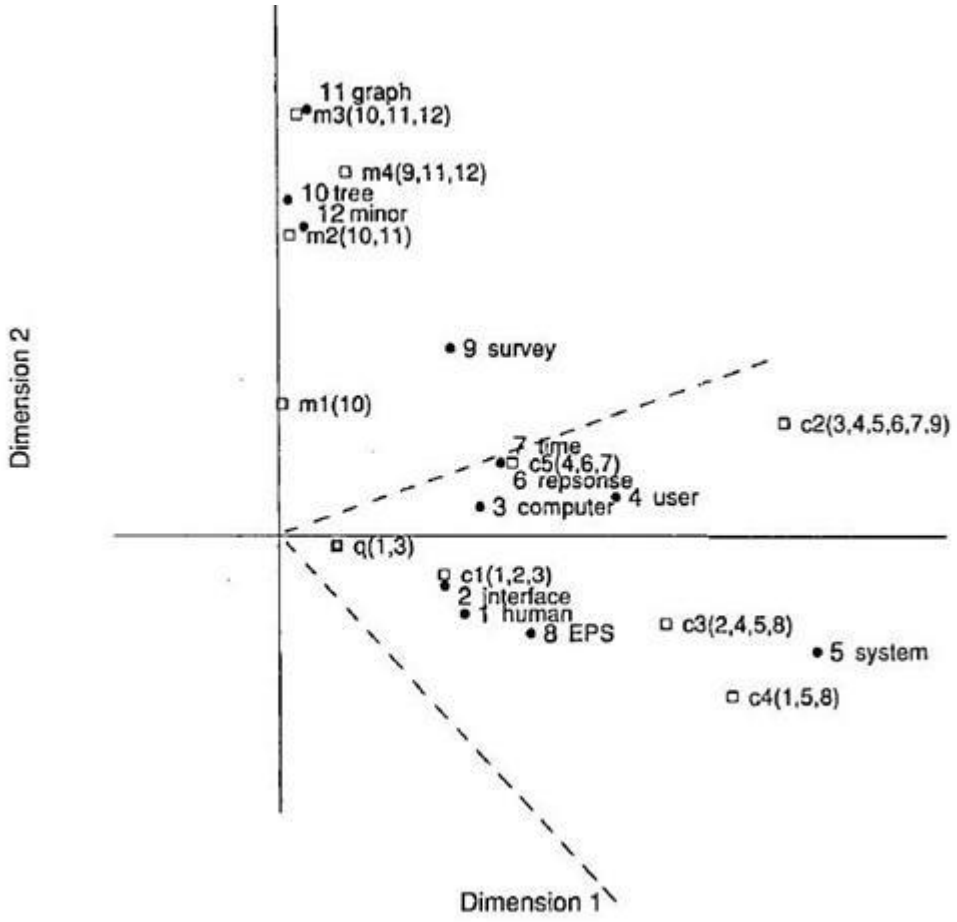

Ilustración 4 Dimensiones LSA tomado [22]

Similitud del coseno, es una función trigonométrica que permite buscar similitud entre documentos y consultas. Para encontrar la coincidencia de la consulta en el espacio reducido del documento de término, la consulta debe transformarse en un documento Psuedo. Los términos están representados en 1 vector. Las funciones de ponderación local y global adecuadas para la colección de documentos se ejecutan en los términos vector. Este vector se compara con todos los vectores de términos y documentos existentes utilizando la similitud de coseno. Los documentos más cercanos a uno, son más similares a la consulta y los documentos más cercanos a cero son menos similares a la consulta. Se devuelve una lista clasificada con todas las similitudes de coseno.

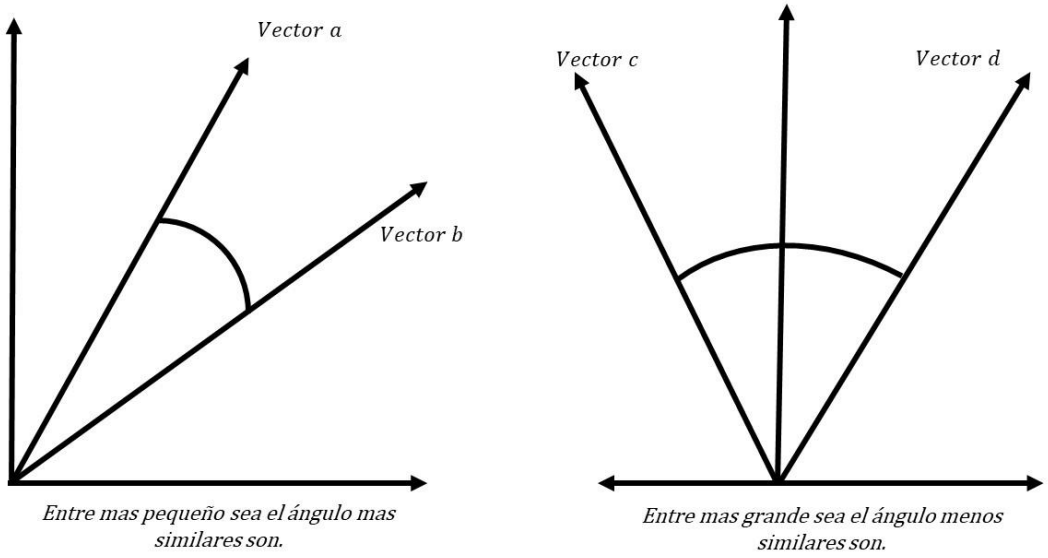

Ilustración 5 Similitud del coseno

3.1.3 TEXTRANK

El algoritmo TEXTRANK (ranking de texto) utiliza diferentes métodos de recuperación de la información para producir las oraciones más relevantes, generando un resumen de acuerdo a las oraciones encontradas. Se basa en gráficos y clasificación de las unidades léxicas de oraciones y palabras usando variaciones del algoritmo PAGERANK (ranking de página), es un algoritmo patentado por la empresa Google muy utilizado por el motor de búsqueda de Google para la relevancia de las páginas web. Se seleccionó porque es independientes del idioma, por lo que no requiere corpus con dominio o anotaciones específicas del idioma[23].

A continuación, se observa el ejemplo para la utilización de TEXTRANK.

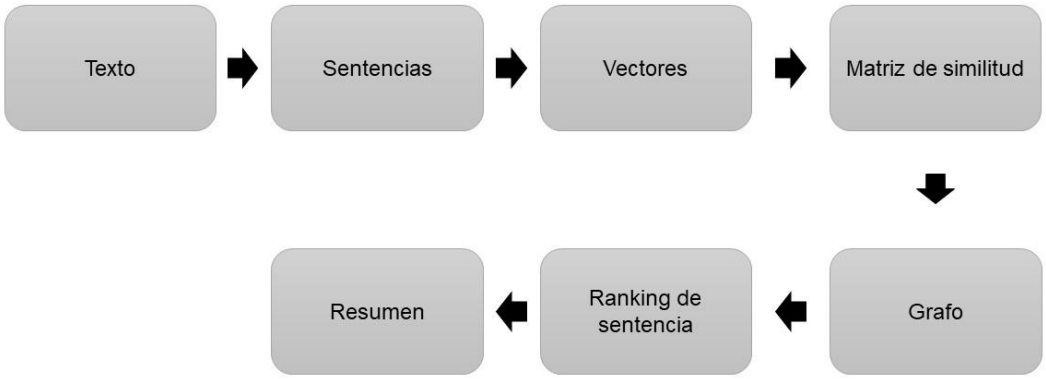

Ilustración 6 Ejemplo TEXTRANK

Los pasos que hace el algoritmo para crear un resumen son:

- **Texto:** Conjunto de enunciados que componen un documento escrito.
- **Sentencias:** En el siguiente paso, encontraremos una representación vectorial (incrustaciones de palabras) para cada oración.
- **Matriz de similitud:** Las similitudes entre los vectores de oraciones se calculan y almacenan en una matriz.
- **Grafo:** La matriz de similitud se convierte luego en una gráfica, con oraciones como vértices y puntajes de similitud como bordes,
- **Ranking de sentencia:** para el cálculo del rango de la oración
- **Resumen:** finalmente, un cierto número de oraciones de primer nivel formando el resumen final.

3.2 ARQUITECTURA

3.2.1 ENTRENAMIENTO DE LA API

Se realizó un API alojada en un servidor web, para poder evaluar el funcionamiento total del sistema, permitiendo que cualquier usuario pueda utilizarla, probarla y evaluarla. La API aplica los algoritmos anteriormente definidos.

La arquitectura detallada en este documento, se realiza a partir de la especificación del estilo “Vista de Módulos”. La vista de Módulos comprende la descripción de las unidades de implementación de software que constituyen la plataforma que será desarrollada. A su vez, se considera también la especificación de la Vista en Capas, la cual describe la organización del código en capas y módulos, detallando cómo las responsabilidades del sistema se distribuyen entre ellos. A continuación, se detalla el diagrama asociado al estilo de vista mencionado.

Este estilo de vista comprende la descripción de las unidades de implementación de software que constituyen la plataforma. En esta sección se considera la especificación de las siguientes vistas: (1) Vista de Descomposición—la cual describe la organización del código en módulos y submódulos, detallando cómo las responsabilidades del sistema se distribuyen entre ellos, y (2) Vista de Modelo de

Datos—a partir de la cual se especifica la estructura estática de la información del dominio, en términos de entidades de datos y sus relaciones.

3.2.2 VISTA DE DESCOMPOSICIÓN.

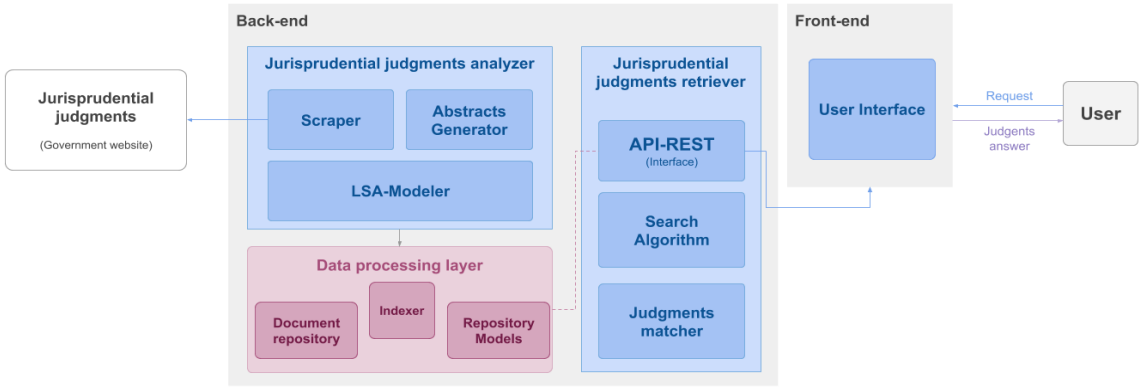

Ilustración 7 Arquitectura del sistema desarrollado

3.2.3 ANALIZADOR DE SENTENCIAS JURISPRUDENCIALES

Este módulo está conformado por tres componentes principales, los cuales se detallan a continuación.

3.2.3.1 SCRAPER

La plataforma utiliza una herramienta web para extraer datos estructurados de enlaces y descripciones de juicios jurisprudenciales publicados en el sitio web, de la corte constitucional de Colombia. Concretamente, este módulo fue desarrollado usando la API de SCRAPY, que es una biblioteca escrita en Python. Utiliza un rastreo que realiza solicitudes y recorre los elementos del sitio web mediante un selector de CSS.

3.2.3.1 GENERADOR DE RESÚMENES

Este módulo crea resúmenes de 300 palabras para cada documento jurisprudencial. Para ayudar al usuario a tener una idea sobre el tema de los documentos. Este módulo se basa en rangos de oraciones de texto usando una variación del TEXTRANK algoritmo. Este algoritmo se basa en gráficos y se seleccionó porque es independiente del idioma, por lo que no requiere corpus con dominio o anotaciones específicas del idioma [23].

3.2.3.2 LSA-MODELER

Este módulo crea tres modelos útiles para el entrenamiento del algoritmo LSA y funcionamiento de la API REST. Es el encargado de procesar, analizar y generar los modelos utilizando el repositorio de documentos Estos modelos son los componentes principales del sistema que permite consultar e indexar las consultas de los usuarios los cuales son: modelo DICCIONARIO, modelo LSA, modelo INDEX.

3.2.4 RECUPERADOR DE SENTENCIAS JURISPRUDENCIALES

Este módulo es el responsable de comprender las consultas del usuario final y transformarlas mediante el modelo LSA, generado a partir de la base de conocimiento de las 28.000 sentencias extraídas, con el fin encontrar las sentencias indexadas que más se aproximen a la consulta del usuario y recuperarlas, para que el usuario pueda consultarlas y seleccionar las que más se ajuste a sus necesidades de información. Para lograr esto, se cuenta con tres módulos principales.

3.2.4.1 API REST

API REST permite que entre sistemas que use HTTP puedan obtener datos o generar operaciones sobre esos datos, en todos los formatos posibles, comúnmente lo más utilizados XML y JSON, permitiendo que diferentes sistemas, sin importar el

lenguaje en que se hicieron pueden compartir datos de una manera simple y eficaz, este módulo recibe la consulta del usuario y retorna el conjunto de sentencias relevantes organizado en orden de relevancia en formato JSON. utilizando un framework llamado Flask escrito en Python.

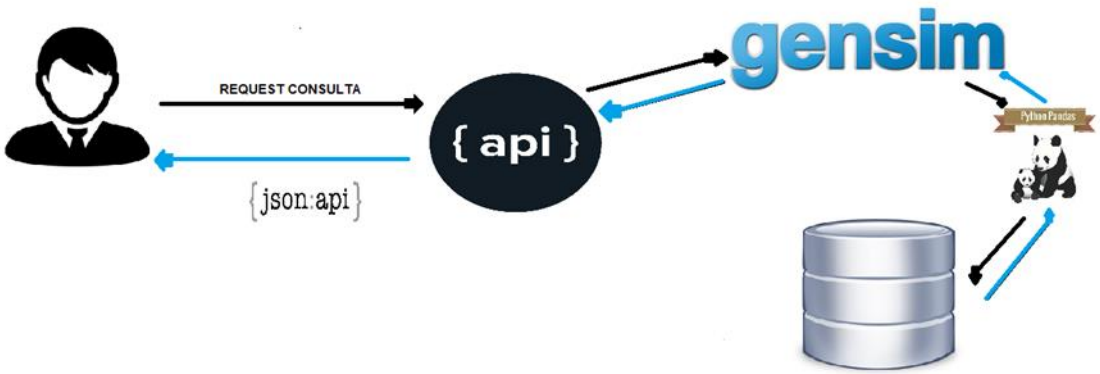

Ilustración 8 Arquitectura de la api

A continuación, se presenta la estructura de respuesta de la API.

- Envió de la consulta

Path	<host>/api/Consulta
Headers	"Content-Type" => " applicationn/json "
Request type	POST

Data
{
“**consulta**”:” tutela”,
“page”: 1,
“**num_reg_page**”: 10,
“**num_temas**”: 1,
“**tipo_algo**”:1,
 	
}
	Consulta: (type: string), tutela a buscar.
	Page: (type: int), pàgina a visitor.
	Num_reg_page: (type: int), número de documentos por página.
	Num_temas: (type: int), se refiere al número de temas que entrenó varia de 1 a 3, el 1 es 100 temas, el 2 200 tema y el 3 300 tema, por defectos son 200 temas.
	Tip_algo: (type: int), el tipo de algoritmo que se quiera cargar.

Body	Content-Type: applicationn/json

Tabla 1 API envió consulta

- Respuesta de la consulta

Código	Mensaje	Descripción

200	{
	"**documentos**":
	{
	“0”: {
	"**texto**": " tutela 1“,
	"**similitud**": 0.90,
	"**link**": "http://www. “,
	"**periodo**": 2016,
	"**titulo**": "T-155-00 “,
	},
	“1”: {
	"**texto**": " tutela 2“,
	"**similitud**": 0.80,
	"**link**": "http://www. “,
	"**periodo**": 2010,
	"**titulo**": "T-123-00 “,
	}
	},
	“**total_registro**”: {28000},
	“**num_registro**”: {10},
	“**pag_actual**”: {0},
	“**pag_sgt**”: {10},
	“**page**”: {1},
	“**time**”: {1}
	}
		Documentos:
		Retorna los documentos más relacionados a consulta. El número de registro depende del parámetro num_reg_page (numero de registro por página).
		
		Total_registro: son los todos los documentos que contiene el dataset.
		
		Page: devuelve la página donde se encuentra la consulta.
		
		Num_registro: devuelve los documentos en contratos, su valor es constante solo cambia en la última página dependiendo si no alcanza a completarel parámetro num_reg_page
		
		Pag_actual y pag_sgt: donde se encuentraen la posición del vector: son para obtener sublista con la siguiente función vector [1:3] permitiendo hacer paginación en un vector
		
		Time: Tiempo que se demoró la consulta

403	Prohibido	El recurso que están tratando de el acceso no es disponible
500	Error	Error en el servidor

Tabla 2 API respuesta consulta

3.2.4.2 EMPAREJADOR DE SENTENCIAS

Dado que las sentencias jurisprudenciales y las consultas del usuario final se encuentran representadas por distribuciones de probabilidad, este componente une el resultado del índex con el repositorio de documentos, devolviendo la información del documento.

3.2.4.3 ALGORITMO DE BÚSQUEDA

Esta modulo recibe las consultas del usuario de parte del módulo API REST y se conecta con otros módulos que son: INDEXER y emparejador de sentencias, para retorna los documentos más relevantes a dicha consulta en formato JSON.

3.2.5 CAPA DE PROCESAMIENTO DE DATOS

Es módulo define las políticas de gestión de datos del sistema, para lo cual se soporta en dos componentes principales.

3.2.5.1 REPOSITORIO DE DOCUMENTOS

Este componente se encarga de almacenar los documentos generados en la extracción de las sentencias jurisprudenciales por parte del SCRAPER. Esto se realiza con el fin de poder generar posteriormente el modelo que representa la estructura semántica oculta de las sentencias, las cuales permitirá comprender la distribución tópicos y así encontrar las sentencias de mayor relevancia a una consulta.

3.2.5.2 INDEXER

El índice de las vistas es una estructura de datos que mejora la velocidad de las operaciones, por medio de identificador único de cada registro, permitiendo un rápido acceso a los registros de las vistas, en este caso, el acceso eficiente a las sentencias relevantes a una consulta determinada. Este módulo se conecta con el modulo repositorio de modelos, para poder indexar los documentos.

3.2.5.3 REPOSITORIO DE MODELOS

Este módulo almacena los tres modelos generados en el modelador LSA; es decir, los términos DICCIONARIO, el modelo LSA y el INDEX.

- **El modelo DICCIONARIO**: tiene guardado todas las palabras tokenizadas con su id y tiene la función de convertir es convertir las consultas del usuario(texto) a un vector de token y ese vector transfórmalo a un vector mapeado si la palabra existe en el diccionario.

- **El modelo LSA**: guarda todas las matrices que fueron creadas utilizando descomposición en valores singulares (SVD), su función es convertir un vector mapeado del modelo diccionario a un vector LSA.

- **El modelo INDEX**: guarda una matriz indexada de documentos. Su función es retorna una matriz de longitud n(documentos) donde su primera columna es el index del documento y la segunda es la similitud que varía entre -1 y 1, respecto a una consulta(texto) que está convertida a un espacio LSA.

3.2.6 INTERFAZ DE USUARIO

La única capa en la interfaz es la interfaz de usuario web. En esta implementación, se desarrolló una aplicación web utilizando las herramientas adecuadas de diseño HTML, que sea sencilla y fácil de usar, puede ser reemplazado por cualquier tipo de aplicación, como aplicaciones de escritorio o móviles.

3.2.7 VISTA DE MODELO DE DATOS

Un modelo de datos describe la estructura estática de la información que se recibe, se genera, se almacena y se entrega en la dinámica de operación de una aplicación, en términos de entidades de datos (o entidades simplemente) y sus relaciones. Éste modelo abstracto comprende los diferentes atributos y sus relaciones, tal como se ilustra en la figura siguiente.

3.2.7.1 PRESENTACIÓN PRINCIPAL

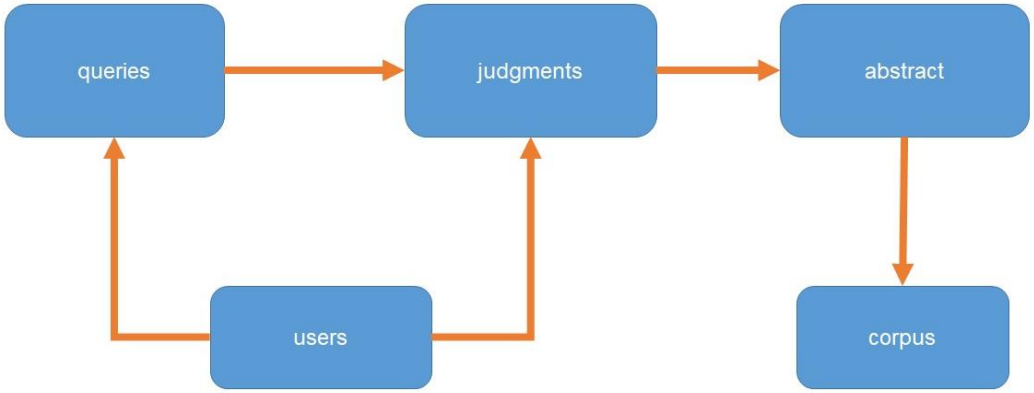

Ilustración 9 Estructura modelo de datos

3.2.7.2 CATÁLOGO DE ELEMENTOS

- **Queries:** esta entidad representa las consultas utilizadas por parte del usuario final para describir su necesidad de información.
- **Judgments:** corresponde a la información de las entidades (id, link, description, abstract_id, periodo).
- **Abstract:** esta entidad contiene el resumen de cada sentencia jurisprudencial.
- **corpus:** es la representación de los Abstract de las sentencias en forma de una matriz documentos-términos.
- **Users:** la entidad que contiene la información de los usuarios que usan el sistema.

3.3 DESARROLLO DEL SISTEMA

En este apartado se definió el algoritmo LSA para las búsquedas de documentos, donde se explicará cada desarrollo, con sus pasos respectivos basado en la arquitectura de la Ilustración 7.

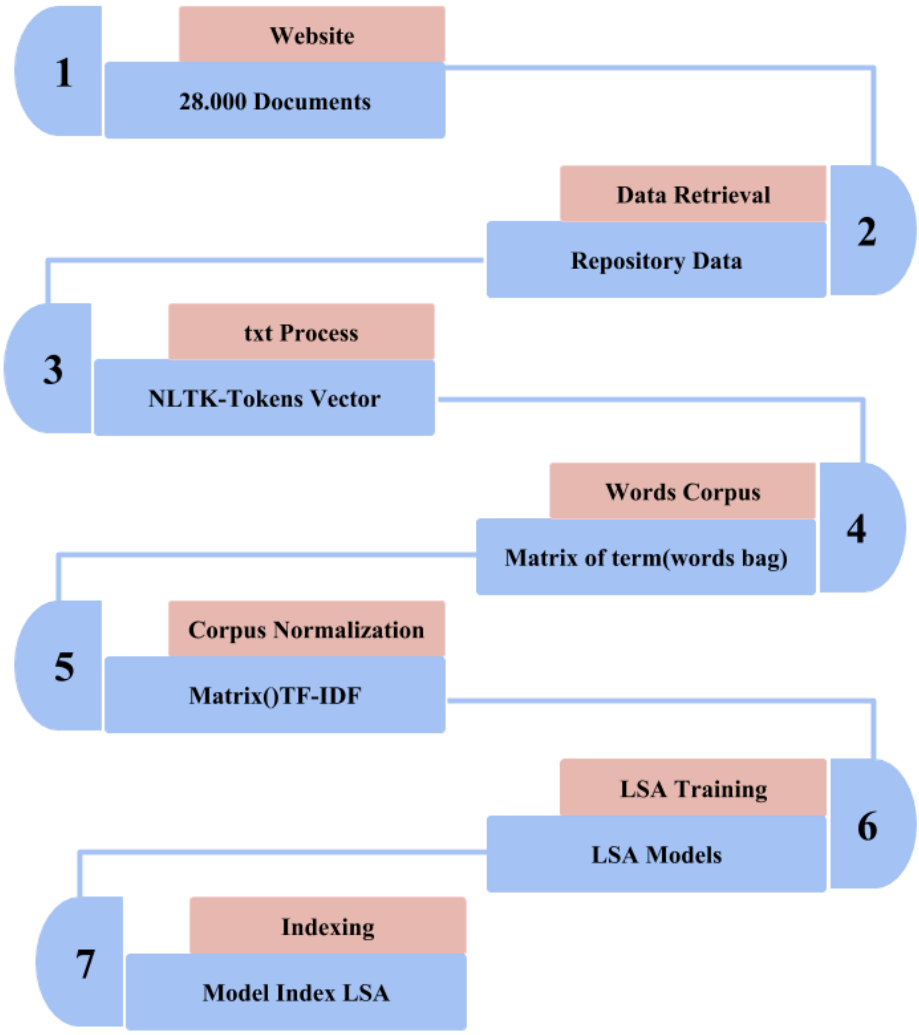

Ilustración 10 Proceso LSA

- **Paso 1:** Con la herramienta Scrapy se procede a capturar todas las tutelas disponible en la página http://www.corteconstitucional.gov.co/relatoria/ [24] que contiene 28000 documentos, se define las etiquetas css en el Scrapy para que las busque y extraiga la información o el link al que deseamos ir.

- **Paso 2:** Cada documento que se encontraron se guarda en el repositorio de documentos, que es el gestor de base de datos MySQL.

- **Paso 3:** Se procede al procesamiento del texto de todos los documentos del repositorio (conjunto de entrenamiento) a un arreglo que contiene solo el texto de cada tutela, luego se recorre cada documento para detectar y eliminar palabras que no aportan ninguna información, luego las convierte a minúsculas, quita los acentos y números luego elimina las palabras repetidas por documento, por último, se tokeniza las palabras resultantes del anterior proceso. Después de hacer todo el procesamiento de la totalidad del arreglo de documentos se crea una matriz donde las filas son los documentos y las columnas son cada palabra(token).

Palabras

Documentos

Doc 1	jurisprudencia	corte	constitucional
Doc 2	violencia	fallo	publico
Doc 3	eps	salud	tecnologia
Doc 3	mujer	drogas	humanos
Doc 4	sector	derechos	indigenas
Doc 5	violacion	violacion	mujeres
Doc 6	mercado	sector	etapas

Ilustración 11 Matriz de texto tokenizada

Luego se crea el modelo DICCIONARIO, se asigna una ID(entero) para cada palabra y crea un diccionario de términos que consiste en un conjunto de tuplas con ID y palabra, esto sirve para un mayor procesamiento a nivel de hardware, ya que el procesamiento de números a texto es más eficiente que el de texto, se guarda este modelo poderlo utilizar después.

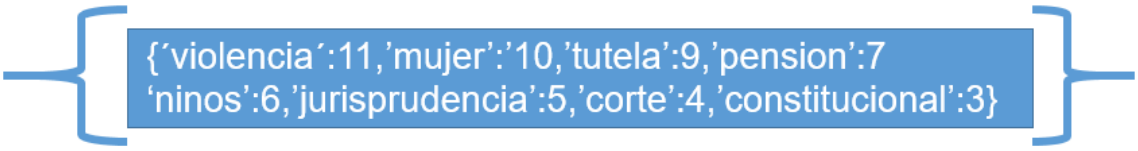

Ilustración 12 Modelo DICCIONARIO

- **Paso 4:** Lo siguiente es crear un corpus (bolsa de palabras) que recibe como parámetro la matriz documentos * palabras(token) y el modulo diccionario. El corpus representa los términos que son las palabras convertidas a token que anteriormente fueron extraídas del procesamiento de texto frente a los documentos, donde una matriz contiene los números que representa cuantas veces la palabra se repite en un documento [25].

Términos

Documentos	Term 1	Term 2	Term 3
Doc 1	1	0	2
Doc 2	3	3	4
Doc 3	2	2	2
Doc 3	1	0	1
Doc 4	3	1	0
Doc 5	4	3	1
Doc 6	2	3	1

Ilustración 13 CORPUS

- **Paso 5:** Una vez que se genera el corpus, se normaliza utilizando el algoritmo TF-IDF se utiliza para normalizar el corpus. Funciona buscando el número de veces que una palabra ha ocurrido en el documento o una frecuencia de palabras en un documento. Su dominio permanece local al documento. La frecuencia de documentos es la fracción de documentos en los que se ha producido la palabra. Se calcula en base a las estadísticas recopiladas de todo el corpus.

Documentos

	Term 1	Term 2	Term 3
Doc 1	0.50	0.10	0.14
Doc 2	0.1	0.01	0.15
Doc 3	0.3	0.31	0.1
Doc 3	0.1	0.3	0.45
Doc 4	0	0.23	0.90
Doc 5	0.90	0.2	0.3
Doc 6	0.10	0.12	0.13

Términos

Ilustración 14 CORPUS TF-IDF

- **Paso 6:** creación del modelo LSA recibe como parámetros el corpus normalizado(tf-idf), el diccionario y el numero k (números de temas). El algoritmo comienza a entrenarse, obteniendo 3 matrices que fueron creadas utilizando las descomposiciones de vectores singulares(SVD) con una dimensionalidad k, estas matrices se guardan en el modelo para que en un futuro se puedan reutilizar. Se guarda este modelo para utilizarlo después.

- **Paso 7:** creación del modelo INDEX recibe como parámetros el corpus normalizado(TF-IDF), el modelo LSA, el modelo diccionario, indexa todos los documentos para hacer consultas de similitud del coseno. Se guarda este modelo poderlo utilizar después.

- Pruebas de consulta de documentos:

  Lo primero es recibir la consulta del usuario(texto) luego limpiar la consulta eliminando palabras que no aportan nada como: palabras vacías, puntuaciones, número y convertirla a minúscula después las convierte en token y con el modelo DICCIONARIO, las mapea a un vector si la palabra se encuentra en el modelo, luego se convierte la consulta al espacio LSA con el modelo LSA, luego el vector LSA pasa al modelo INDEX, donde se busca los documentos más similares mediante la similitud del coseno retornado un

vector de longitud n documentos donde la primera fila es el index del documento y la segunda columna es la similitud que varía entre -1 y 1, luego se ordena de mayor a menor el vector por la columna similitud, el módulo de emparejador de sentencias recibe el vector y recorre el vector obteniendo el index y busca en el módulo repositorios de documentos para retornar el documento correspondiente al index, por último, se formatea a JSON.

CAPÍTULO 4

4. CONSTRUCCIÓN DE LA BASE DE DATOS.

4.1 CONSTRUCION DE LA BASE DE DATOS

Para crear la base de datos donde se almacenan los documentos que posteriormente serán procesados por el sistema para ser visualizados, se establecen las siguientes herramientas para la creación de la base de datos de documentos. Se utilizaron herramientas como Scrapy, y el gestor de base de datos MySQL a continuación se representa la construcción de la base de datos.

Como parte inicial para la construcción de la base de datos se utilizó Scrapy una herramienta de Python que permite inspeccionar etiquetas HTML de las páginas de internet donde su información se almacena en las diferentes etiquetas del fron-end, de esta manera se ingresa al repositorio de la corte constitucional http://www.corteconstitucional.gov.co/relatoria/ [24], donde se almacena la mayoría de documentos, descargando a la base de datos cerca de 28000 documentos con los campos de id, link, nombre, resumen, descripción, ver ilustración 12.

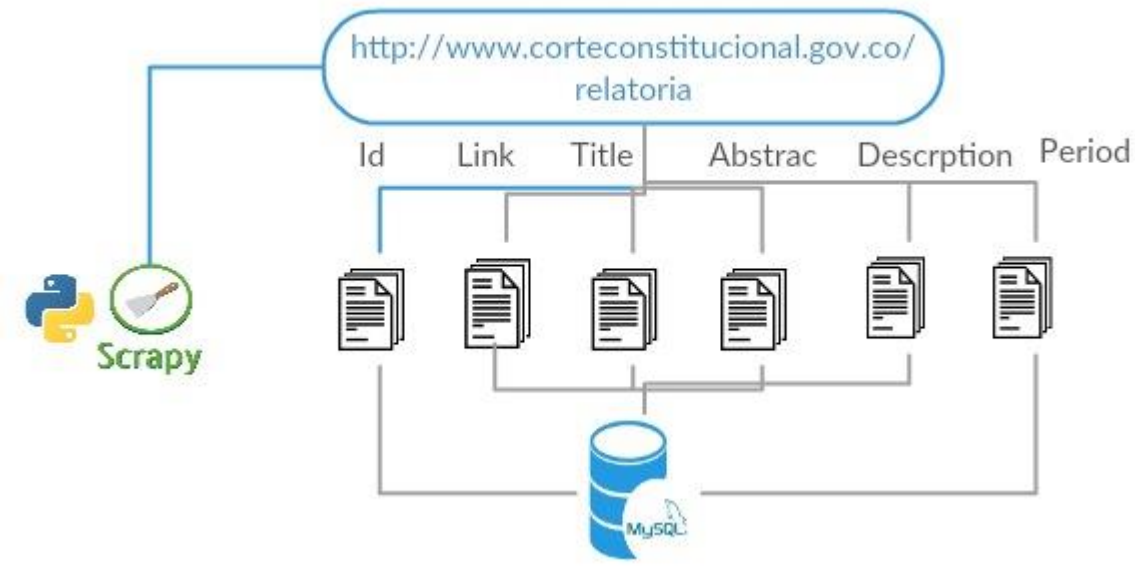

Ilustración 15 SCRAPY procesamiento

Para la construcción se utilizó esta página http://www.corteconstitucional.gov.co/relatoria/, porque es libre y permite acceder fácilmente sin login(iniciar sesión) y sin ninguna restricción.

Los campos elegidos para guardar la información son los que ofrecen la página y eran los más importantes para el usuario que se pueden identificar:

- **Id:** es un identificador para la base de datos, no ex visible para el usuario.
- **Url**: link del PDF, que se utilizaría para que el usuario pueda descargar el usuario.
- **Periodo:** para que el usuario supiera en que año se aceptó la tutela
- **Título:** comprende el nombre de la tutela.
- **Resumen:** el resumen no es sacado de la página, para ello se utilizó el algoritmo TEXTRANK que se explicó anteriormente.
- **Descripción:** contiene todo el texto del documento.

La herramienta SCRAPY se procede a enviar múltiples arañas que no salen del dominio especifico de la página http://www.corteconstitucional.gov.co/relatoria/ el cual contiene 28000 documentos, se define las etiquetas CSS en el SCRAPY para que las busque y extraiga la información o el link al que deseamos ir.

Paso 1: la herramienta SCRAPY envía una araña a la página principal, donde recorre todos los links de tutela, enviando otras arañas por cada link, que se muestran en la siguiente ilustración.

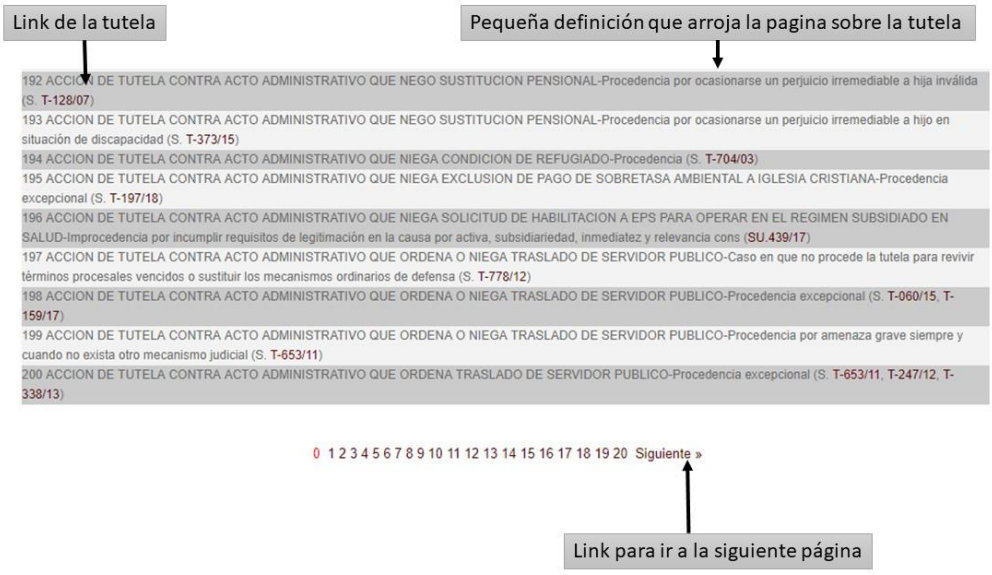

Ilustración 16 Pagina inicial SCRAPY

Cuando termina la primera página sigue con la posterior por medio de link siguiente, si no hay más se detiene.

Paso 2: Cuando las arañas se dirigen al link de la tutela capturan los siguientes campos por las etiquetas CSS.

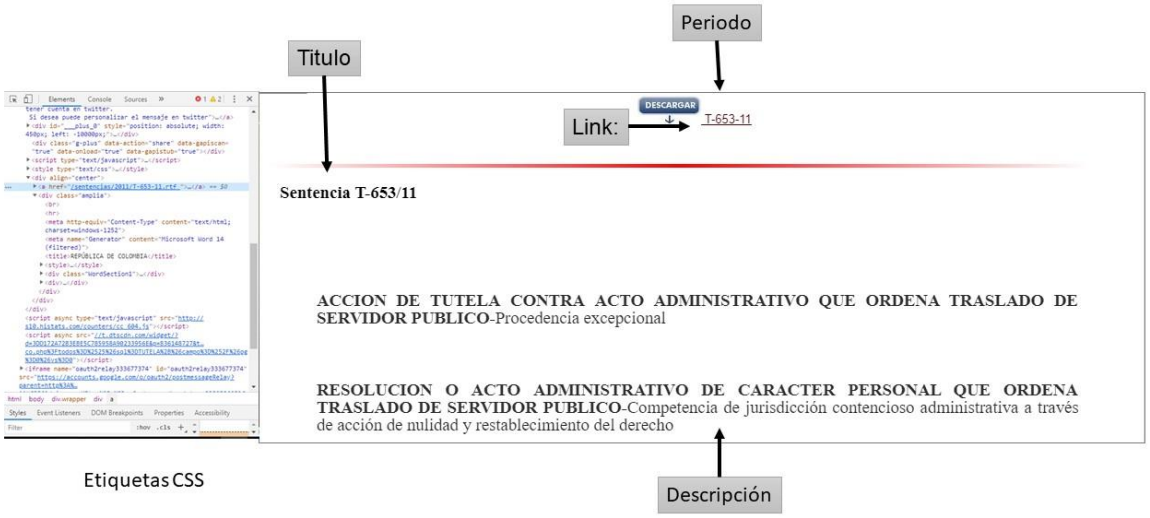

Ilustración 17 Pagina del documento de la tutela

Paso 3: Cada vez que una araña termina se guarda el documento en la base de datos MySQL con los siguientes campos mencionados anteriormente. Para la descripción se utilizó BEAUTIFUL SOUP para extraer todo el texto eliminando estilos CSS y script JavaScript.

A continuación, se da una definición de las herramientas que se utilizaron.

4.1.1 MYSQL

Es un gestor de bases de datos relacional muy popular frente a Oracle y Microsoft SQL SERVER, se utiliza mucho en el desarrollo web y tiene una comunidad muy activa, ofreciendo escalabilidad y actualizaciones muy rápidas, a problemas que se encuentre. MYSQL permite transacciones simultaneas muy potentes, permitiendo que varios usuarios puedan conectarse a ella y realicen operaciones. Esto se puede realizar debido a que este gestor de base de datos utiliza multadillos mediante el kernel, tiene dos licencias una gratuita y otra comercial.

4.1.2 SCRAPER

Es una herramienta para en rastrear páginas web y buscar contenido específico en ellas, dando la posibilidad de crear data set para guardarlo en archivos planos, csv, Excel y bases de dato. Su función en este proyecto es de conectarse a un sitio web que contiene una gran cantidad de sentencias jurisprudenciales (Cerca de 28000), sobre el cual se encarga de extraer toda la información de la sentencia y almacenarla en el Repositorio de Documentos.

4.1.3 SCRAPY

Es una biblioteca escrita en Python. Utiliza un rastreo que realiza solicitudes y recorre los elementos del sitio web mediante un selector de CSS.

4.1.4 BEAUTIFUL SOUP

Es una biblioteca escrita en Python. Sirve para extraer datos de HTML Y XML. Muy utilizada en el SCRAPER ya que analizadores de etiquetas CSS muy potentes y limpieza de etiquetas que no deseamos capturar.

CAPÍTULO 5

5. EVALUACION DE ALGORITMOS DE INTELIGENCIA ARTIFICIAL PARA LA INDEXACIÓN DE DOCUMENTOS

5.1 INSTRUMENTO DE VALIDACIÓN

Esta sección describe los experimentos desarrollados para evaluar la satisfacción del usuario con el sistema. Los usuarios trabajaban e interactúan con el sistema y luego proporcionaron sus reflexiones sobre su grado de satisfacción alcanzado en diferentes aspectos. Por otro lado, se integra un cuestionario que permita evaluar a los usuarios, la plataforma de identificación recomendación y búsqueda de precedentes judiciales, este se basa en el modelo propuesto en [26] para evaluar el éxito del Sistema de recuperación de documentos basado en jurisprudenciales , este modelo debido a las numerosas similitudes con el caso en mano, por lo tanto, debido a su facilidad de aplicación con una baja necesidad de adaptación. Además, desde el punto de vista del usuario es más fácil para estos proporcionar información si hay pocas preguntas, considerando que los usuarios por lo general son reacios a responder encuestas. En particular, se plantearon 3 preguntas con una escala de respuesta por el usuario.

· **Q1**. ¿La plataforma genera precisión a la hora de encontrar una sentencia? En este caso nos centramos en observar si la plataforma brinda los resultados esperados de las búsquedas.
· **Q2**. ¿La plataforma genera una descripción (resumen) adecuada a la búsqueda realizada? En este caso nos centramos en observar si los resúmenes son eficientes.
· **Q3**. ¿Crees que la plataforma genera un resultado ágil (tiempo de respuesta) por cada búsqueda? La idea es medir la reacción de los usuarios, y el tiempo medido por el usuario en respuesta a la búsqueda realizada por la plataforma.

A continuación, se observa los resultados del primer instrumento de validación.

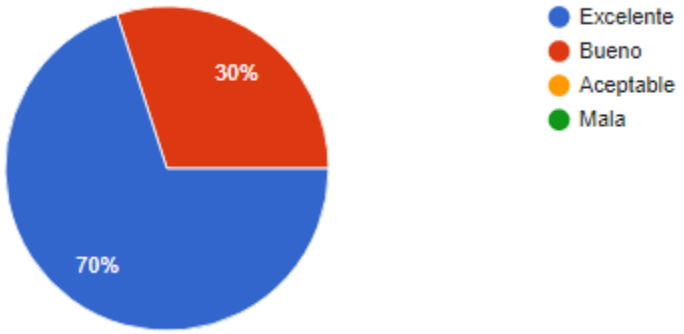

Ilustración 18 ¿La plataforma genera una DESCRIPCIÓN (Resumen) adecuada a la búsqueda realizada?

Para la pregunta 2 ver ilustración 18 con un 70% de los encuestados mencionan que el buscador genera resúmenes de manera excelente esto quiere decir que el algoritmo implementado genera confianza cuando genera el resumen para la visualización de la descripción para los usuarios además el 30% restante mencionan que el resumen generado es bueno esto es favorable ya que ninguno de los usuarios piensa que los resúmenes generados por parte de la aplicación son malos para describir los casos de jurisprudencia.

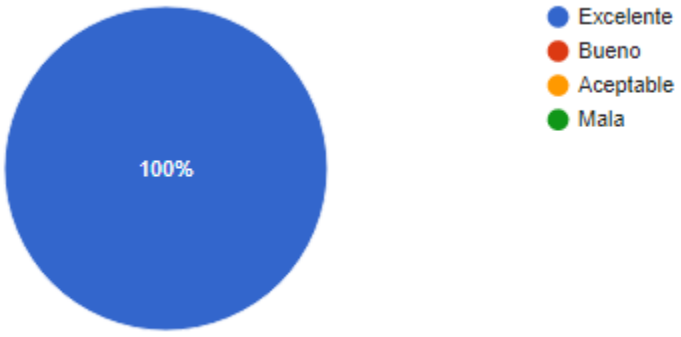

Ilustración 19 ¿Crees que la plataforma genera un resultado ágil (tiempo de respuesta) por cada búsqueda realizada?

Para la pregunta 3 ver ilustración 19 con un 100% de los encuestados mencionan que el buscador genera un tiempo de respuesta para cada búsqueda de manera

excelente, esto significa que los algoritmos implementados, las tecnologías y servidor genera confianza cuando genera una búsqueda donde el grado de satisfacción en tiempo de respuesta es muy alto.

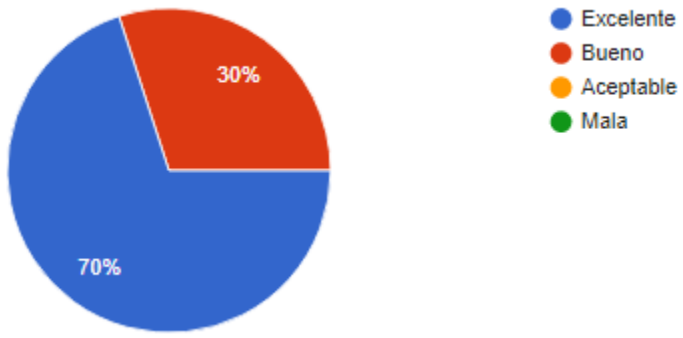

Ilustración 20 ¿La plataforma genera precisión a la hora de encontrar una sentencia?

Para la pregunta 1 ver ilustración 20 con un 70% de los encuestados mencionan que el buscador genera precisión excelente en el momento de realizar una búsqueda, en otros términos cuando el usuario busca diferentes sentencias estos encuentran de manera favorable además se encuentran los diferentes documentos que ayudan a resolver sus casos en particular, por otra parte el 30% restante mencionan que la precisión del buscador es buena esto contribuye a demostrar que la aplicación genera búsquedas precisas y aproximadas a hora de realizar una búsqueda.

5.2 TEST DE USABILIDAD DEL SISTEMA

Para la evaluación de la plataforma de identificación recomendación y búsqueda de precedentes judiciales, se utilizó un método empírico de evaluación: prueba de usuario, donde usuarios finales pudieron interactuar y probar la plataforma, en ella se le realizo una serie de preguntas sobre la experiencia de uso de la plataforma

utilizando la escala de usabilidad del sistema (SUS), los usuarios calificaron de Muy de acuerdo a Muy en desacuerdo los siguientes ítems:

- Creo que me gustaría utilizar esta aplicación con frecuencia.
- Me parece que la aplicación es innecesariamente compleja.
- En mi opinión la aplicación me pareció fácil de usar.
- Creo que necesitaría ayuda de un técnico experto para poder utilizar la aplicación.
- Me parece que las distintas funciones en la aplicación fueron bien integradas.
- Pienso que la aplicación tiene muchas inconsistencias
- Creo que la mayoría de personas aprenderían a utilizar esta aplicación muy rápidamente
- Me parece muy complicada de usar esta aplicación.
- Me sentí muy cómodo usando la aplicación.
- Necesité aprender muchas cosas antes de que pudiese manejar esta aplicación.

La sesión fue realizada por diferentes 10 usuarios entre ellos estudiantes de derecho, abogados, y diferentes usuarios del ámbito legal en Colombia dado que, según Jakob Nielsen las pruebas permiten encontrar casi la misma cantidad de problemas de usabilidad que se encontrarían utilizando muchos más participantes en las pruebas.

A cada participante se le asignó un computador para acceder a la plataforma, donde resolvió el cuestionario. En esta sesión participaron tres (3) monitores quienes fueron los encargados de dirigir la prueba y registrar cualquier pregunta adicional que realicen los participantes.

En esta evaluación se plantearon los siguientes casos dados por expertos en el tema:

1. un paciente diagnosticado con deficiencia renal en ambos riñones, que se encuentra afiliado a una EPS que funciona en la ciudad de Popayán, hizo la solicitud para el trasplante de riñón (que era urgente y necesario para su salud), teniendo algunos posibles donantes familiares que estaban dispuestos a realizarse los exámenes. En Popayán se dilató el trámite durante un año, con la excusa de que no había convenios, hasta que fue remitido a la ciudad de Bogotá, en esta ciudad le hicieron iniciar el protocolo completo de nuevo y llevaba varios meses a espera de fecha para la cirugía, pues su hermana ya había salido compatible. Su abogado interpone tutela en Popayán para que sea resuelto con prontitud el caso?

2. Un informante avisa a las autoridades que en un predio se encuentra un presunto miembro de la guerrilla de las FARC-EP. Al día siguiente se realiza un allanamiento al predio, donde ingresa la policía de forma violenta tumbando la cerca, y disparando gases lacrimógenos a la vivienda, donde se encuentra el joven que es capturado, esposado, y arrojado boca abajo, la audiencia de legalización de captura se realiza dentro del término legal, y es retenido en las instalaciones de la policía durante 3 días, al cabo de los cuales lo dejan en libertad aduciéndole que fue confundido con una persona homónima de su mismo nombre y apellidos.

Para la pregunta 1 ¿Encontraste documentos que ayudaron a resolver estos casos?

Ilustración 21 Documentos encontrados por usuarios

El 100% de los usuarios encuestados respondieron que, si encontraron los documentos para poder resolver los casos previamente mencionados ver Ilustración 21, esto demuestra que entre el repositorio de datos existen documentos necesarios que satisfacen las búsquedas de los usuarios.

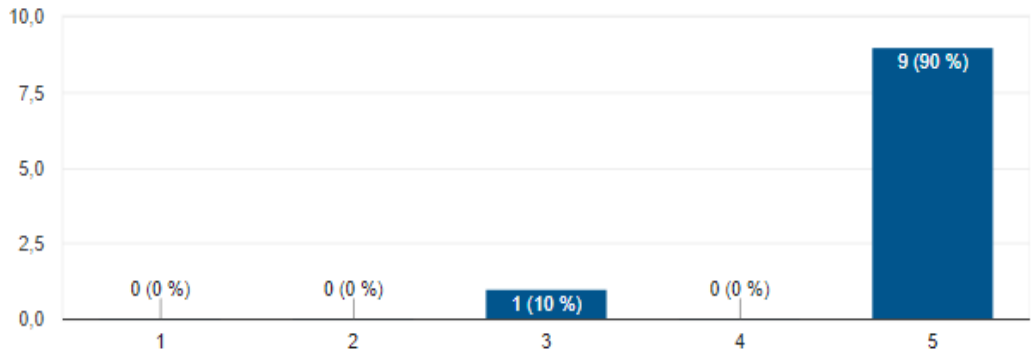

Ilustración 22 Creo que me gustaría utilizar esta aplicación con frecuencia

Para la pregunta 2 ver ilustración 22 los usuarios con un 90% mencionan estar muy de acuerdo en que les gustaría utilizar esta aplicación con más frecuencia, esto es muy importante debido a que en Colombia no se cuenta con herramientas libres para la búsqueda ágil de documentos de jurisprudencia, por otra parte esto quiere decir también que los componentes están bien integrados y que por tano el funcionamiento es óptimo para la utilización de los usuarios, el 10% restante califa con un 3 esto quiere decir que en la escala de 1 a 5 menciona no estar ni muy de acuerdo pero tampoco muy desacuerdo.

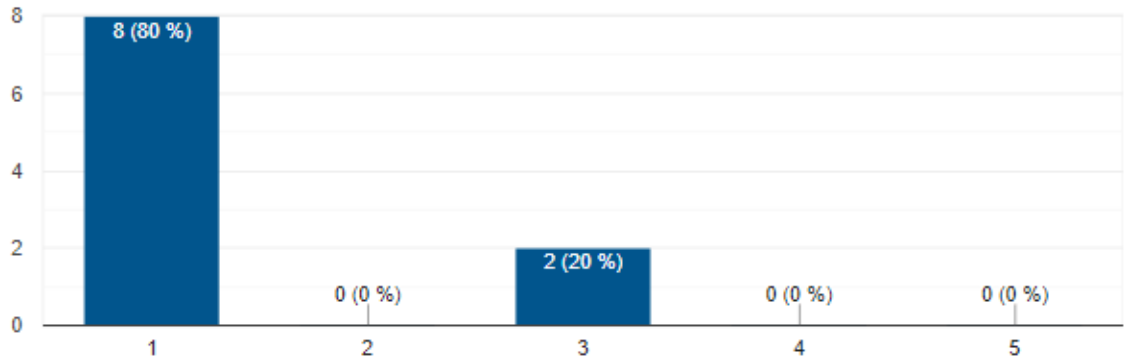

Ilustración 23 Me parece que la aplicación es innecesariamente compleja

Para la pregunta 3 ver ilustración 23 con un 80% de los encuestados mencionan estar muy desacuerdo en el sentido de que la aplicación es fácil de usar, por otra esto quiere decir que para cualquier usuario experto en derecho y demás personas del común les sería posible utilizar este tipo de herramientas ya que la tendencia de esta pregunta demuestra que les parece una herramienta idónea para realizar sus búsquedas, el 20% restante tiene tendencia a estar muy deacuerdo ya que en la escala de 1 a 5 donde 1 está muy desacuerdo y 5 muy deacuerdo los usuarios demuestran estar entre una escala intermedia demostrando que el buscador no genera complejidad de funcionalidad.

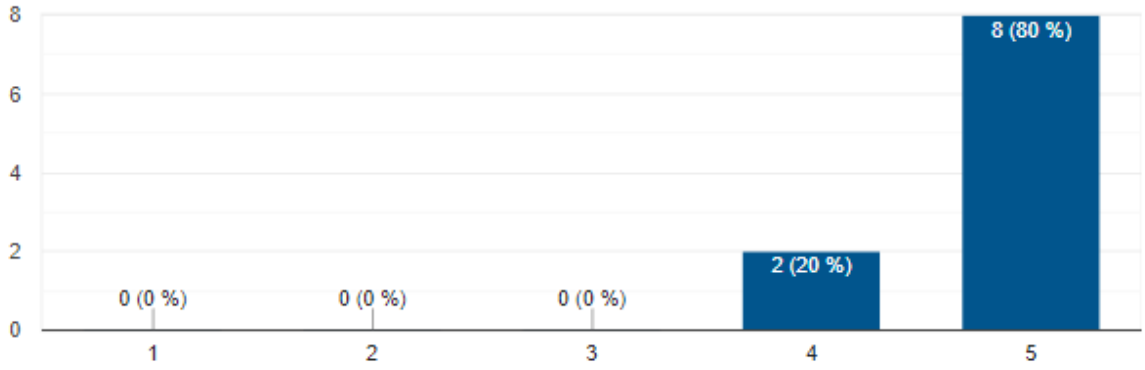

Ilustración 24 En mi opinión la aplicación me pareció fácil de usar

Para la pregunta 4 ver ilustración 24 con un 80% de los encuestados mencionan estar muy de acuerdo en el sentido de que la aplicación es fácil de usar, esto quiere decir que para cualquier usuario les sería viable utilizar este tipo de herramientas de búsqueda además la tendencia de esta pregunta demuestra que es herramienta idónea para realizar sus búsquedas debido a que el 20% restante tiene tendencia a estar muy deacuerdo donde en la escala de 1 a 5 donde 1 está muy desacuerdo y 5 muy deacuerdo los usuarios demuestran estar entre una escala favorable demostrando que el buscador para los usuarios les genera confianza al ser fácil de usar.

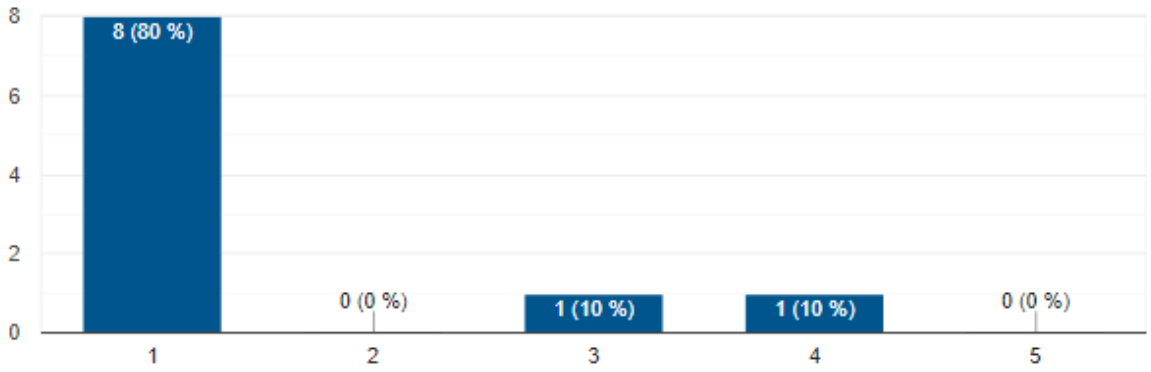

Ilustración 25 Creo que necesitaría ayuda de un técnico experto para poder utilizar la aplicación

Para la pregunta 5 ver ilustración 25 con un 80% de los encuestados mencionan estar muy desacuerdo en sentido de que necesitarían ayuda de un técnico o experto para poder utilizar la o realizar una búsqueda en la aplicación, esto quiere decir que para la mayoría de usuarios se sentiría en la capacidad de realizar una búsqueda. Por otra parte 10 % de la población encuestada demuestra estar entre una escala intermedia y el 10% restante necesitaría la ayuda de un técnico para poder llevar a cabo una búsqueda dentro del sistema.

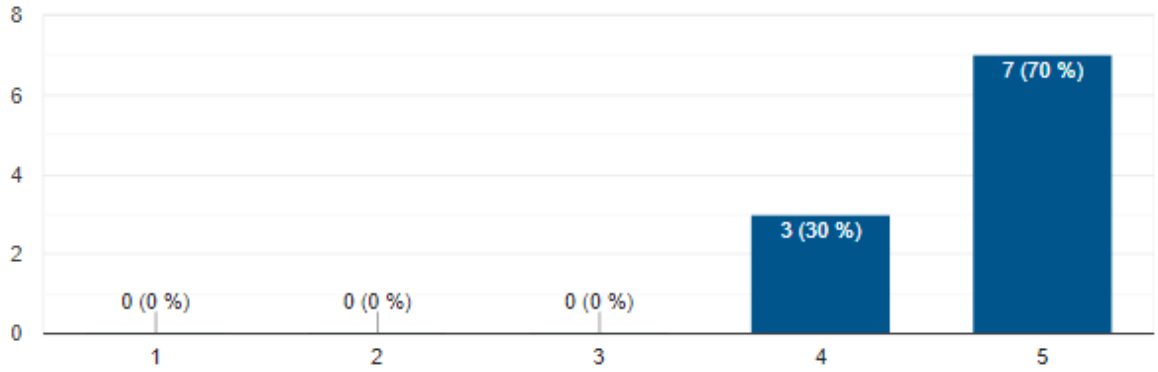

Ilustración 26 Me parece que las distintas funciones en la aplicación fueron bien integradas

Para la pregunta 6 ver ilustración 26 con un 70% de los encuestados mencionan estar muy desacuerdo con la funcionalidad del sistema esto quiere decir que las pruebas realizadas con anterioridad para integrar cada uno de los componentes del buscador fueron exitosas además de estar bien integradas, el 30% restante mantiene una escala de favorabilidad esto quiere decir que un 100% de los encuestados están conformes con cada uno de los componentes integrados dentro de la aplicación.

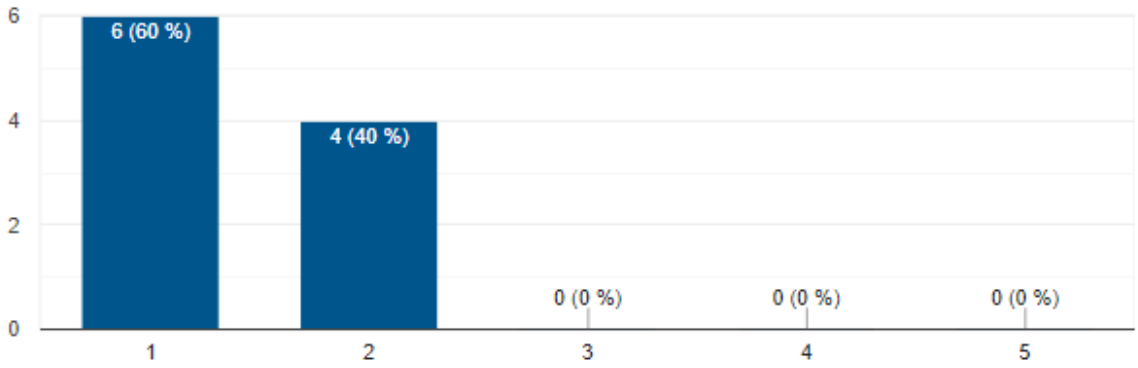

Ilustración 27 Pienso que la aplicación tiene muchas inconsistencias

Para la pregunta 7 ver ilustración 27 el 60% de los encuestados mencionan estar muy desacuerdo, esto se debe a que la aplicación genera confianza a la hora de realizar una búsqueda, donde los usuarios encuestados no observaron ninguna inconsistencia dentro de la aplicación durante su uso. Por otra parte 40 % de la población encuestada demuestra estar entre una escala favorable donde la

tendencia es que el sistema genera confianza a la hora de realizar una búsqueda ya está responderá a las necesidades del usuario.

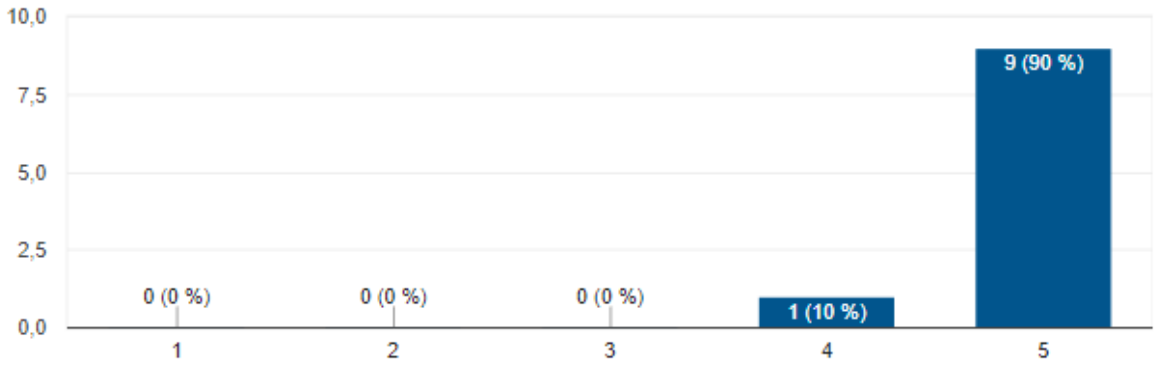

Ilustración 28 Creo que la mayoría de personas aprenderían a utilizar esta aplicación muy rápidamente

Para la pregunta 8 ver ilustración 28 con un 90% de los encuestados mencionan estar muy de acuerdo con la funcionalidad del sistema por ello creen que la mayoría de personas, usuarios del sistema aprenderían de forma rápida a utilizar la aplicación, el 10% restante mantiene una escala de favorabilidad esto quiere decir que un 100% de los encuestados creen que cualquier usuario aprendería de fácil manera a generar búsqueda de jurisprudencia dentro de la aplicación.

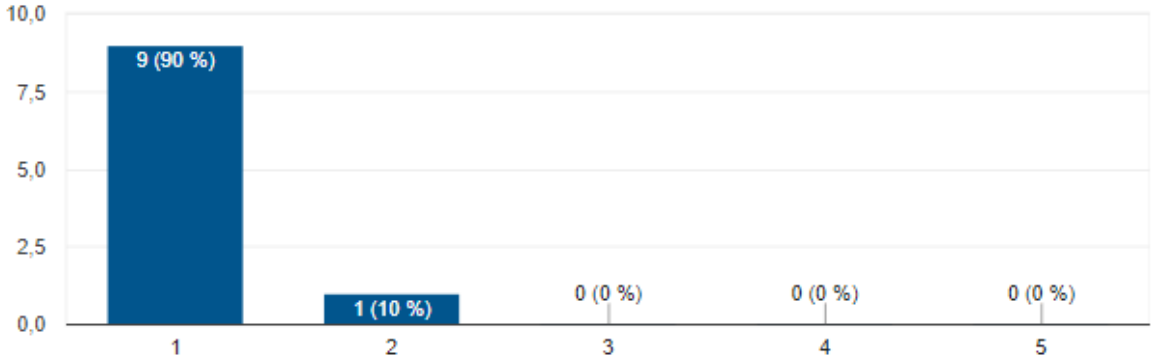

Ilustración 29 Me parece muy complicada de usar esta aplicación.

Para la pregunta 9 ver ilustración 29 con un 90% de los encuestados mencionan estar muy desacuerdo en el sentido de que la aplicación es complicada de usar, esto quiere decir que para cualquier usuario experto en derecho y demás personas del común les sería posible utilizar este tipo de herramientas ya que la tendencia de esta pregunta demuestra que les parece una herramienta idónea para realizar sus búsquedas, el 10% restante tiene tendencia a estar muy desacuerdo ya que en la escala de 1 a 5 donde 1 está muy desacuerdo y 5 muy deacuerdo los usuarios demuestran estar entre una escala favorable demostrando que el buscador no genera complejidad de funcionalidad

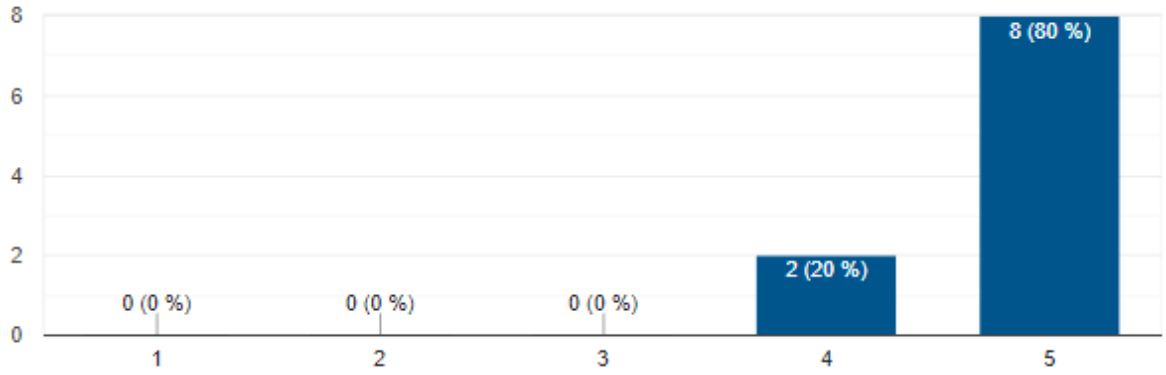

Ilustración 30 Me sentí muy cómodo usando la aplicación

Para la pregunta 10 ver ilustración 30 con un 80% de los encuestados mencionan estar muy de acuerdo con el sistema ya que se sienten cómodos haciendo uso de esta esto quiere decir que las pruebas realizadas con anterioridad para integrar cada uno de los componentes del buscador fueron exitosas además de estar bien integradas, el 20% restante mantiene una escala de favorabilidad esto quiere decir que un 100% de los encuestados están conformes al sentirse cómodos desarrollando búsquedas dentro de la aplicación.

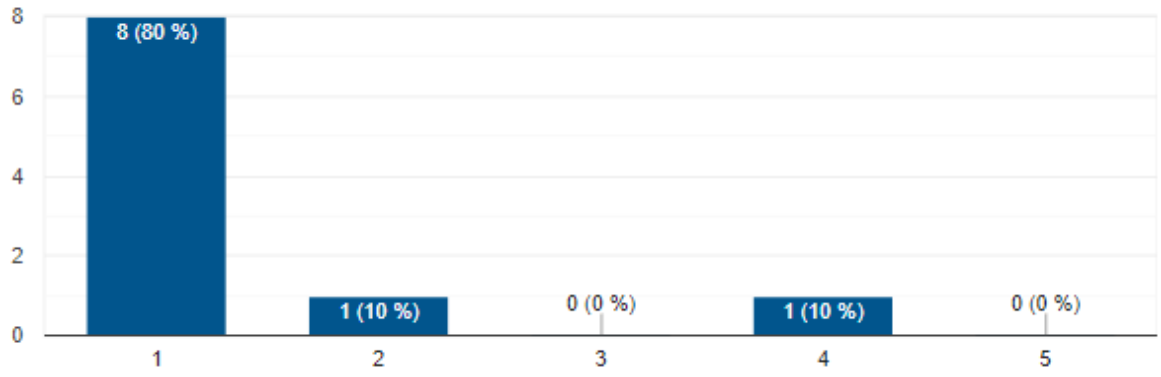

Ilustración 31 Necesité aprender muchas cosas antes de que pudiese manejar esta aplicación

Para la pregunta 11 ver ilustración 31 con un 90% de los encuestados mencionan estar muy desacuerdo en el sentido de que la aplicación es complicada donde necesitaron aprender muchas cosas para poder usar la aplicación, esto quiere decir que para cualquier usuario experto en derecho y demás personas del común les sería posible utilizar este tipo de herramientas ya que la tendencia de esta pregunta demuestra que les parece una herramienta idónea para realizar sus búsquedas, el 10% restante tiene tendencia a estar muy desacuerdo esto es favorable ya se mantiene una tendencia descrita anteriormente, por otra parte el 10% restante mencionan que necesitarían aprender algunas cosas para poder realizar una búsqueda dentro del sistema.

CAPÍTULO 6

6. CONCLUCIONES Y TRABAJO FUTURO

En conclusión, se determinó que el sistema de respuesta a las búsquedas es eficiente y entre más palabras claves el usuario ingrese, más preciso serán resultados generados por el buscador.

Se identificó que las tutelas en este caso la constitucional, arrojo la búsqueda realizada por los usuarios que están dentro del campo del derecho estuvieron satisfechos, por que podían encontrar las sentencias adecuadas que concordaran con sus casos.

Se alcanzó un mayor interés parte de los usuarios para utilizar el sistema de indexación, evidenciado en las encuestas realizadas debido a su fácil adherencia y comprensión.

Se debe destacar que esta herramienta solo tenía las tutelas de la corte constitucional, ya que existe la penal entre otras. La comparación entre el buscador realizado se hizo con la página de la corte constitucional. No se comparó con otras plataformas donde se debe pagar debido a que estas plataformas tienen una base de datos más robusta, por consiguiente, el resultado en algunas tutelas seria deficiente.

Como trabajos futuros se desea trabajar con todas las tutelas, disponibles de Colombia, para un mejor resultado ya que la base del conocimiento se amplía encontrado un espacio semántico más amplio y preciso. Con el fin de evaluarlo con sistema de pagos más robusto como VLEX, VLEX es una plataforma de consultas que utiliza inteligencia artificial para mejorar sus consultas y posee todas las tutelas de Colombia.

CAPITULO 7

7. BIBLIOGRAFIA

[1] Colciencias, “AGENDA ESTRATÉGICA DE INNOVACIÓN - NODO JUSTICIA,” *Colciencias*, pp. 1–30, 2014.

[2] CARLOS ANDRÉS CÁRDENAS GÓEZ, “UNIFICACIÓN JURISPRUDENCIAL VERSUS EL PRESEDENTE JUDICIAL,” *Univ. St. TOMAS*, pp. 1–12, 2014.

[3] U. Jurisprudencial, H. La, and J. P. Sarmiento-erazo, “EL RECURSO EXTRAORDINARIO DE EN LO CONTENCIOSO-ADMINISTRATIVO ? unification remedy : Towards,” *Univ. los Andes*, pp. 247–282, 2011.

[4] A. Wyner, R. Mochales-palau, M. Moens, and D. Milward, “Approaches to Text Mining Arguments from Legal Cases,” *Univ. Coll. London*, pp. 60–79, 2010.

[5] G. Venturi, “Legal Language and Legal Knowledge Management Applications,” pp. 3–4, 2010.

[6] K. L. Vester and M. C. Martiny, “INFORMATION RETRIEVAL IN DOCUMENT SPACES USING CLUSTERING Department,” *Tech. Univ. denmark*, pp. 1–266, 2005.

[7] A. Gelbukh, “Procesamiento de Lenguaje Natural y sus Aplicaciones,” vol. I, pp. 1–6, 2010.

[8] V. Parra, “DERECHO COMPARADO,” *Univ. la Gran Colomb.*, vol. 4, pp. 241–264, 2004.

[9] J. T. Valdés, *Derecho informatico*. 2008.

[10] A. P. Rodríguez and R. Artículo, “El Precedente en el Derecho Colombiano Un Estudio Comparado con la Jurisprudencia Alberto Poveda Rodríguez 1,” *Univ. Católica Colomb.*, pp. 1–28, 2010.

[11] E. Muelas, “Gestión de la Información : organización , búsqueda y recuperación en Internet,” *Fundec*, pp. 1–49, 2017.

[12] C. T. López and L. A. García, “Textual representation in semantic vector space,” *Rev. Cuba. Ciencias Informáticas*, vol. 10, no. 2, pp. 148–180, 2016.

[13] L. Ania and T. Pombert, “El uso de los buscadores en Internet,” *Scielo*, pp. 1–

13, 2003.

[14] F. M. Santiago, M. Ángel, G. Cumbreras, A. M. Ráez, and M. C. Díaz, "Procesamiento del Lenguaje Natural, Revista nº 53, septiembre de 2014," *Asoc. Lingüística Comput.*, pp. 1–216, 2014.

[15] A. Pinto, H. G. Oliveira, and A. O. Alves, "Comparing the Performance of Different NLP Toolkits in Formal and Social Media Text ∗," *Found. Sci. Technol.*, no. 3, pp. 1–3, 2014.

[16] E. Fersini and F. Sartori, "Improving the Effectiveness of Multimedia Summarization of Judicial Debates through Ontological Query Expansion," 2011, vol. 83, pp. 450–463.

[17] L. Ot, D. C. Furquim, and L. Vera, "Clustering and Categorization of Brazilian Portuguese Legal Documents," *Ed. Springer Berlin Heidelb.*, pp. 272–273, 2012.

[18] M. Hern, "Analisis automatico de textos en espanol utilizando NLTK por," *Univ. LA LAGUNA Anal.*, pp. 1–47, 2016.

[19] A. A. Hakim, A. Erwin, K. I. Eng, M. Galinium, and W. Muliady, "Automated Document Classification for News Article in Bahasa Indonesia based on Term Frequency Inverse Document Frequency (TF-IDF) Approach," *Int. Conf. Inf. Technol. Electr. Eng.*, pp. 1–4, 2014.

[20] M. P. Kherwa and P. Bansal, "Latent Semantic Analysis : An Approach to U nderstand Semantic of Text," *2017 Int. Conf. Curr. Trends Comput. Electr. Electron. Commun.*, pp. 870–874, 2017.

[21] R. María, A. Semántico, and L. Teoría, "Análisis Semántico Latente : ¿ Teoría psicológica del significado ? Rosa María Gutiérrez *," *Rev. Signos ISSN*, pp. 303–323, 2005.

[22] M. Claudia, "Revisión y aplicación de aspectos de la lingüística matemática de la información La recuperación de información en el siglo XX Revisión y aplicación de aspectos de modelización matemática de la Tesina presentada para optar al grado de Licenciado por :," *Mem. acadèmica*, pp. 1–135, 2008.

[23] F. Barrios, F. López, L. Argerich, and R. Wachenchauzer, "Variations of the Similarity Function of TextRank for Automated Summarization," *Univ. Buenos*

Aires, pp. 1–8, 2016.

[24] C. Constitucional, “Corte contitucional,” *Ministerio de Justicia y del Derecho Colombia*, 2015. .

[25] G. De Jorge-Botana, “La técnica del Análisis de la Semántica Latente (LSA/LSI) como modelo informático de la comprensión del texto y el discurso,” *Univ. Auton. MADRID*, pp. 1–447, 2010.

[26] N. Manouselis, H. Drachsler, R. Vuorikari, H. Hummel, and R. Koper, “Recommender Systems in Technology Enhanced Learning,” in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer US, 2011, pp. 387–415.