Departamento:

Departamento de Ingeniería e Investigaciones Tecnológicas

Programa de acreditación:

PROINCE

Programa de Investigación[1]:

Código del Proyecto:

C241

Título del proyecto

Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado

PIDC: ☐

Elija un elemento.

PII: ☐

Elija un elemento.

Director:

SPOSITTO, Osvaldo Mario

Codirector:

RYCKEBOER, Hugo

Integrantes:

LEDESMA, Viviana
BOSSERO, Julio César
GARGANO, Cecilia
MATTEO; Lorena
MORENO, Edgardo
PROCOPIO, Gastón
SAIZAR, Victoria
MACIAS, Patricio
CONTI, Laura

Asesor- Especialista:

GARCÍA, Sergio
PEREZ VILLAR, Gustavo

Alumnos de grado: (Aclarar si tiene Beca UNLaM/CIN)

OJEDA, Juan
QUINTANA, Fabio

Resolución Rectoral de acreditación: N°

445/21

Fecha de inicio:

01/01/2021

Fecha de finalización:

31/12/2022

[1] Los Programas de Investigación de la UNLaM están acreditados con resolución rectoral, según lo indica la Resolución HCS Nº 014/15 sobre Lineamientos generales para el establecimiento, desarrollo y gestión de Programas de Investigación a desarrollarse en la Universidad Nacional de La Matanza. Consultar en el departamento académico correspondiente la inscripción del proyecto en un Programa acreditado.

A. Desarrollo del proyecto (adjuntar el protocolo)

A.1. Grado de ejecución de los objetivos inicialmente planteados, modificaciones o ampliaciones u obstáculos encontrados para su realización (desarrolle en no más de dos (2) páginas)
Los objetivos planteados para el proyecto de investigación fueron los siguientes:

a) Desarrollar e implementar un front-end Web del SRI basado en el prototipo desarrollado previamente por el propio equipo en el proyecto C151.

b) Paralelizar el proceso Indexación Semántica Latente (ISL) implementando una arquitectura híbrida (CPU-GPU).

c) Construir un corpus de documentos jurídicos.

d) Evaluar el tiempo de respuesta y la pertinencia de los documentos recuperados.

Durante la primera etapa del proyecto se llevaron adelante una serie de subtareas de investigación que sirven de apoyo a las tareas planificadas de armado del corpus y de análisis del resultado del proceso de lematización. Si bien en dichas etapas es necesaria la intervención manual por parte de los expertos, se buscó unificar términos similares en los documentos jurídicos a fin de reducir la dimensionalidad (cantidad de términos) del corpus, simplificando así las tareas manuales de revisión por parte de los usuarios. Para ello se aplicaron diversas técnicas automáticas, como ser aplicación de Expresiones Regulares (ER), distancia de Hamming y de Levenshtein (para encontrar similitud de términos), funciones de deduplicación y limpieza de datos, entre otras.

Respecto al proceso de ISL, se profundizó en la utilización de ER para búsqueda y reemplazo de cadenas de textos dentro de textos más extensos. Una ER es una secuencia de caracteres que define un patrón de búsqueda sobre un texto. Así también se revisaron trabajos respecto a la utilización del reconocimiento de Entidades Nombradas (EN), para ser empleadas juntamente con las ER. Se trabajó en una propuesta orientada a la incorporación de datos en formato fecha y de dichas EN (Leyes, Acordadas, Decretos, Artículos, etc.), utilizando la librería para ER que trae C# (REGEX). Todo esto de cara a la especialización del Sistema de Recuperación de Información para un contexto jurídico.

Con relación a lo anterior, esta parte del trabajo ha quedado parcialmente volcado en un artículo que se redactó cuyo título es *"Propuesta para la construcción de un Corpus Jurídico utilizando Expresiones Regulares"*, que fue aceptado para su publicación en el congreso CACIC 2021. Este tema resultó de mucho interés y fue seleccionado para su inclusión en un capítulo del libro "Computer Science – CACIC 2021" de la serie Communications in Computer and Information Science" (CCIS) de la editorial Springer. El mismo, titulado "Lexical Analysis Using Regular Expressions for Information Retrieval from a Legal Corpus", fué publicado en 2022.

En esta misma línea, con el fin de reducir ese costo de intervención manual y, para mejorar la performance en la búsqueda exhaustiva de patrones realizada inicialmente mediante el uso de ER, se buscó aplicar técnicas complementarias que podrían ayudar a reducir la dimensionalidad. Para ello se realizó un estudio para la búsqueda del mejor umbral de coincidencia que surge de aplicar medidas de similitud léxica a los términos resultantes del proceso de indización y organización del corpus. Aunque este procedimiento no exime de la necesidad de contar con el experto humano, puede ser

de gran ayuda para minimizar el esfuerzo implicado en su trabajo, debido a que permite acotar el volumen de términos en los que debe centrarse para elegir aquel que mejor represente a la EN. Los principales resultados de este estudio se reflejan en el artículo *"Determinación del umbral inferior de coincidencia aplicando medidas de edición a términos jurídicos"*, publicado a la revista REDDI, en diciembre de 2022.

Por otra parte, ampliando lo planificado inicialmente se transcribió la lógica del algoritmo de Snowball utilizado en el proyecto anterior (2014) del lenguaje C# a C para mayor portabilidad. Se construyeron varias funciones de String no previstas por el lenguaje C, como la obtención de cadenas de caracteres y la conversión de acentos en vocales. Se prepararon dos lotes de prueba con 27.266 y 1.113.014 términos respectivamente. A estos lotes, se los pasó por el proceso de Lematización que ofrece el sitio de https://snowballstem.org/. Sumado a esto, se corrigieron lemas en los tiempos verbales. Para ello fue necesario modificar el algoritmo de Snowball, de este modo, una vez que el proceso obtiene el lema, este es buscado en un lote de verbos almacenados, de encontrase, es reemplazado por el lema correcto. Resultados iniciales de las pruebas realizadas durante esta actividad fueron publicadas en el trabajo "*Implementación de un lematizador para la lengua española*" presentado en CONAIISI 2021. Se termino de lematizar el segundo lote de palabras, este se está incorporando al programa para realizar distintas modificaciones en el orden que se ejecutan los pasos.
Adicionalmente, se realizó un estudio preliminar, en el que se investigó la localización de documentos en espacios métricos utilizando el algoritmo K-means. Se realizaron experimentos con distintas configuraciones de dimensionalidad y normas de distancia. Una ampliación de esta actividad se ha publicado en el artículo "Resultados preliminares de una técnica de localización de documentos en espacios métricos utilizando K-means", presentado en el congreso CONAIISI 2022.
Con respecto al armado del corpus documental jurídico, se gestionó las correspondientes autorizaciones y la Suprema Corte de Justicia de la Provincia de Buenos Aires, proveyó un archivo de 23 Gigabyte que incluía más de 300 mil documentos del fuero civil y comercial, separados por distintas "etiquetas". Para procesar este archivo, se desarrolló un programa en C# que permite segmentar y armar a los documentos que conformarán el corpus jurídico.
Además, se desarrolló el SRI web que incluye configurar la carpeta donde se encuentran los archivos a indexar. Una vez configurada, se inicia el proceso de indexación, cuya duración depende del tamaño del corpus. Luego, relacionado a las búsquedas, se ingresan las palabras clave a buscar, el sistema utiliza la última indexación realizada para encontrar los documentos que coincidan con las mismas. El sistema devuelve los resultados en un listado y ofrece la posibilidad de verlos o descargarlos.
En términos de implementación, se utilizó un algoritmo de paralelización para una arquitectura híbrida CPU-GPU. Sin embargo, debido a retrasos en la entrega de la documentación necesaria para construir el corpus por parte de la Corte Suprema, no se pudieron realizar pruebas exhaustivas para evaluar la eficiencia, el tiempo de respuesta y la relevancia de los documentos recuperados. Como resultado, no se cumplió el objetivo e) establecido inicialmente durante el período de vigencia del proyecto, no obstante ello estas pruebas se realizarán en el segundo semestre del año en curso y los resultados serán presentados en la WICC o CACIC 2024.

B. Principales resultados de la investigación

B.1. Publicaciones en revistas (informar cada producción por separado)

Artículo 1:	
Autores	*Matteo, Lorena; Ledesma Viviana; Spositto Osvaldo*
Título del artículo	*Determinación del umbral inferior de coincidencia aplicando medidas de edición a términos jurídicos*
N° de fascículo	*2*
N° de Volumen	*7*
Revista	*REDDI*
Año	*2022*
Institución editora de la revista	*ULaM - DIIT*
País de procedencia de institución editora	*Argentina*
Arbitraje	SI
ISSN:	*2525-1333*
URL de descarga del artículo	*https://reddi.unlam.edu.ar/index.php/ReDDi/article/view/188*
N° DOI	https://doi.org/10.54789/reddi.7.2.4

B.2. Libros

Libro 1	
Autores	
Título del Libro	
Año	
Editorial	
Lugar de impresión	
Arbitraje	Elija un elemento.
ISBN:	
URL de descarga del libro	
N° DOI	

B.3. Capítulos de libros

Autores	Spositto, Osvaldo; Bossero, Julio; Moreno, Edgardo; Ledesma, Viviana; Matteo, Lorena
Título del Capitulo	*Lexical Analysis Using Regular Expressions for Information Retrieval from a Legal Corpus*
Título del Libro	Computer Science-CACIC 2021 Publicado en Springer Nature Switzerland
Año	*2022*
Editores del libro/Compiladores	P. Pesado and G. Gil (Eds.): CACIC 2021
Lugar de impresión	*Suiza*
Arbitraje	SI
ISBN:	*978-3-031-05902-5*
URL de descarga del capítulo	*https://link.springer.com/chapter/10.1007/978-3-031-05903-2_21*
N° DOI	

B.4. Trabajos presentados a congresos y/o seminarios

B.4.1	
Autores	Spositto, Osvaldo; Ryckeboer, Hugo; Ledesma, Viviana; Procopio, Gastón; Matteo, Lorena; Gargano, Cecilia; Bossero, Julio; Moreno, Edgardo; Saizar, Victoria; Macias, Patricio; Ojeda, Juan; Quintana, Fabio; Conti, Laura; García, Sergio; Perez Villar, Gustavo
Título	*Propuesta para la construcción de un Corpus Jurídico utilizando Expresiones Regulares*
Año	*2021*
Evento	XXVII Congreso Argentino de Ciencias de la Computación (CACIC 2021)
Lugar de realización	Virtual
Fecha de presentación de la ponencia	4 al 8 de octubre de 2021
Entidad que organiza	Facultad de Ciencias Exactas de la Universidad Nacional de Salta
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	*https://cacic2021.unsa.edu.ar/wp-content/uploads/2021/11/LIBRO-DE-ACTAS-CACIC-2021-SALTA.pdf*

B.4.2	
Autores	Bossero, Julio, Spositto, Osvaldo; Ledesma, Viviana Ryckeboer, Hugo; Conti, Laura; García, Sergio; Moreno, Edgardo; Matteo, Lorena; Saizar, Victoria; Macias, Patricio; Quintana, Fabio; Perez Villar, Gustavo; Gargano, Cecilia; Procopio, Gaston
Título	*Implementación de un lematizador para la lengua española*
Año	*2021*
Evento	9º Congreso Nacional de Ingeniería Informática / Sistemas de Información (CoNaIISI 2021)
Lugar de realización	Virtual
Fecha de presentación de la ponencia	05 de noviembre de 2021
Entidad que organiza	Universidad Tecnológica Nacional Facultad Regional Mendoza
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	*https://www4.frm.utn.edu.ar/conaiisi/*

B.4.3	
Autores	Spositto, Osvaldo; Ryckeboer, Hugo; Bossero, Julio; Moreno, Edgardo; Ledesma, Viviana; Procopio, Gastón; Matteo, Lorena; Gargano, Cecilia; Saizar, Victoria; Macias, Patricio; Quintana, Fabio; Ojeda, Juan; Conti, Laura; García, Sergio; Pérez Villar, Gustavo
Título	*Adecuación de un sistema de recuperación de información para su utilización en un contexto jurídico*
Año	*2022*
Evento	XXIV Workshop de Investigadores en Ciencias de la Computación (WICC 2022)
Lugar de realización	Virtual
Fecha de presentación de la ponencia	abril de 2022
Entidad que organiza	Universidad Champagnat. Mendoza
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	*http://sedici.unlp.edu.ar/handle/10915/144389*

B.4.4	
Autores	Spositto, Osvaldo; Ryckeboer, Hugo; Bossero, Julio; Moreno, Edgardo; Ledesma, Viviana; Procopio, Gastón; Matteo, Lorena; Gargano, Cecilia; Saizar, Victoria; Macias, Patricio; Quintana, Fabio; Ojeda, Juan; Conti, Laura; García, Sergio; Pérez Villar, Gustavo
Título	*Resultados preliminares de una técnica de localización de documentos en espacios métricos utilizando K-means.*
Año	*2022*
Evento	10º Congreso Nacional de Ingeniería Informática / Sistemas de Información (CoNaIISI 2022)
Lugar de realización	Facultad Regional Concepción del Uruguay de la Universidad Tecnológica Nacional, Entre Ríos.
Fecha de presentación de la ponencia	noviembre de 2022
Entidad que organiza	Red de Carreras de Ingeniería Informática / Sistemas de Información -RIISIC- perteneciente al CONFEDI
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	*https://rtyc.utn.edu.ar/index.php/ajea/article/view/1146/1059*

B.5. Otras publicaciones

Autores	
Año	
Título	
Medio de Publicación	

C. Otros resultados. Indicar aquellos resultados pasibles de ser protegidos a través de instrumentos de propiedad intelectual, como patentes, derechos de autor, derechos de obtentor, etc. y desarrollos que no pueden ser protegidos por instrumentos de propiedad intelectual, como las tecnologías organizacionales y otros. Complete un cuadro por cada uno de estos dos tipos de productos.

C.1. Títulos de propiedad intelectual. Indicar: Tipo (marcas, patentes, modelos y diseños, la transferencia tecnológica) de desarrollo o producto, Titular, Fecha de solicitud, Fecha de otorgamiento

Tipo	Titular	Fecha de Solicitud	Fecha de Emisión

C.2. Otros desarrollos no pasibles de ser protegidos por títulos de propiedad intelectual. Indicar: Producto y Descripción.

Producto	Descripción

D. Formación de recursos humanos. Trabajos finales de graduación, tesis de grado y posgrado. Completar un cuadro por cada uno de los trabajos generados en el marco del proyecto.

D.1. Tesis de grado

Director (apellido y nombre)	Autor (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis

D.2 Trabajo Final de Especialización

Director (apellido y nombre)	Autor (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título del Trabajo Final

D.2. Tesis de posgrado: Maestría

Director (apellido y nombre)	Tesista (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis
Bossero Julio	Casuccelli Mauro	UNLaM		En curso	Estudio comparativo de DBSCAN, KMEANS con redes neuronales en un Sistema de Recuperación de Información

D.3. Tesis de posgrado: Doctorado

Director (apellido y nombre)	Tesista (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis
Palma Luis María	Conti Laura	UNLaM		En curso	Implementación de la Inteligencia Artificial y su Regulación en los Procesos de Gestión en la Ejecución Penal en la Provincia de Buenos Aires

D.4. Trabajos de Posdoctorado

Director (apellido y nombre)	Posdoctorando (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título del trabajo	Publicación

E. Otros recursos humanos en formación: estudiantes/ investigadores (grado/posgrado/ posdoctorado)

Apellido y nombre del Recurso Humano	Tipo	Institución	Período (desde/hasta)	Actividad asignada[2]
CONTI, Laura	Estudiante de Doctorado	UNLaM	1/1/2021 – 31/12/2022	- Colaboración en el armado del corpus documental. - Pruebas del SRI
GARCÍA, Sergio	Estudiante de Doctorado	UNLaM	1/1/2021 – 31/12/2022	- Colaboración en el armado del corpus documental. - Pruebas del SRI
QUINTANA, Fabio	Estudiante de Grado	UNLaM	1/1/2021 – 31/12/2022	- Implementación de la paralelización de la ISL - Desarrollo de SRI web
OJEDA Juan	Estudiante de Grado	UNLaM	1/1/2021 – 31/12/2022	- Implementación de la paralelización de la ISL - Desarrollo de SRI web

F. Vinculación[3]: Indicar conformación de redes, intercambio científico, etc. con otros grupos de investigación; con el ámbito productivo o con entidades públicas. Desarrolle en no más de dos (2) páginas.

- Este proyecto de investigación se conforma por un equipo de trabajo multidisciplinario. Como se mencionó previamente, se ha trabajado en colaboración con responsables del Juzgado de Ejecución Penal Nro. 2 de Morón, a cargo de la Dra. Laura Conti. Así también, se cuenta con la participación del Prosecretario de la Subsecretaría de Tecnología Informática de la Suprema Corte de Justicia de Bs. As. Ambos, por su parte, aportarán los documentos pertinentes para el armado del corpus jurídico que servirá de base para las pruebas del SRI. Se espera que los resultados de esta investigación sean de

[2] Descripción de la/s actividad/es a cargo (máximo 30 palabras)

[3] Entendemos por acciones de "vinculación" aquellas que tienen por objetivo dar respuesta a problemas, generando la creación de productos o servicios innovadores y confeccionados "a medida" de sus contrapartes.

interés y puedan ser utilizados en futuros desarrollos para la Suprema Corte de Justicia de la Provincia de Buenos Aires.

- Se estableció contacto y se realizaron reuniones de intercambio con un equipo de investigadores de la Facultad de Ciencias Jurídicas de la Universidad Nacional de la Plata. El equipo sigue una línea de investigación sobre Derecho y Tecnología llamado GECSI (Grupo de Estudios de la Complejidad en la Sociedad de la Información). La lectura de distintos artículos que hemos publicado despertó el interés de GECSI en este proyecto, no solo para un intercambio recíproco de conocimientos, sino también pensando en organizar y llevar adelante a futuro algunas actividades académicas conjuntas, lo cual puede resultar sumamente enriquecedor para ambos equipos.
- El 11 de octubre de 2022 se llevó adelante una reunión de intercambio con el Ministerio de Justicia de Perú. Los participantes por parte de nuestro equipo de investigación fueron: Osvaldo Sposito, Viviana Ledesma, Laura Conti y Gustavo Pérez Villar. Por la otra parte, el Juez Supremo, Magistrado Héctor Lama More, quien lidera el Centro de Investigaciones del Poder Judicial de Perú junto a algunos de sus colaboradores, entre ellos, el Gerente de Informática del ministerio. Fue una reunión muy productiva, se dieron a conocer los avances que se están realizando en Perú con relación a la incorporación de nuevas tecnologías en los procesos judiciales y algunos obstáculos con los que se han encontrado. Por nuestra parte, se explicó el estado de situación de la justicia digital, en especial en la Provincia de Buenos Aires, además se comentaron las líneas de investigación en las que está trabajando el equipo, los avances alcanzados en el proyecto Experticia y los beneficios potenciales que se esperan a partir de su implementación. El Magistrado manifestó su interés en avanzar con este tipo de intercambios y conocer más de nuestro trabajo de cara a evaluar la posibilidad de su aplicación en la justicia de Perú.

- Se estableció contacto con el Laboratorio de Inteligencia Artificial para la Fiscalía de Estado (FEPBA IALab) de la Provincia de Buenos Aires, el 20 de abril de 2023 fuimos convocados a una reunión de intercambio de experiencias. Participaron de la reunión, por parte de la fiscalía, Mariano Cervellini y Miguel Carbone y, de nuestro equipo, Osvaldo Spossito y Viviana Ledesma. En su caso, ellos están trabajando en el desarrollo de Velox, un prototipo para la gestión interna de la fiscalía, comentaron las técnicas de IA y de recuperación de información que están aplicando. Mostraron interés en colaborar con nuestro equipo de investigación debido a las publicaciones relacionadas con el SRI y la Experticia que han estado siguiendo. Siendo que se trata de un laboratorio recientemente creado en la Fiscalía de Estado les gustaría contar para sus proyectos con nuestra experiencia y colaboración. Las líneas de comunicación han quedado abiertas para avanzar en esta relación de colaboración reciproca a futuro.

G. Otra información. Incluir toda otra información que se considere pertinente.

Actividades de Difusión en Eventos Científicos:

- Presentación Video: Ledesma y Bossero. Título de Artículo: "*Propuesta para la construcción de un Corpus Jurídico utilizando Expresiones Regulares"*. Evento: XXVII Congreso Argentino de Ciencias de la Computación (CACIC 2021). Organizado por: Facultad de Ciencias Exactas de la Universidad Nacional de Salta. Fecha: octubre de 2021.
- Expositor: Bossero. Título de Artículo *"Implementación de un lematizador para la lengua española".* Evento: CONAIISI 2021, 9º Congreso Nacional de Ingeniería Informática / Sistemas de Información. Organizado por: Universidad Tecnológica Nacional Facultad Regional Mendoza. Fecha: noviembre de 2021.
- Expositor: Bossero. Título del trabajo presentado: *Implementación de un lematizador para la lengua española*. Evento: CONAIISI 2022. Organizado Facultad Regional Concepción del Uruguay de la UTN. Fecha: noviembre de 2022.
- Presentación Video: Ledesma y Bossero participaron del XXIV Workshop de Investigadores en Ciencias de la Computación – WICC 2022 – con el articulo "*Adecuación de un Sistema de Recuperación de Información para su Utilización en un Contexto Jurídico*" y su correspondiente video y poster. Organizado por la Universidad Champagnat. Fecha: abril de 2022.
- Ledesma, Procopio y Pérez Villar participaron como panelistas en las 51 JAIIO Jornadas Argentinas de Informática. Conferencia de cierre del SID - Simposio Argentino de Informática y Derecho. Título de la conferencia: "*EXPERTICIA y la necesidad de adaptar la gestión judicial a la evolución tecnológica*". Organizado por la Sociedad Argentina de Informática (SADIO). Fecha: octubre de 2022.
- Expositor: Pérez Villar. Título de la conferencia: Innovación y mejora del servicio público de justicia en Argentina. Evento: III Congreso Internacional Expediente Judicial Electrónico del Poder Judicial del Perú. Organizado por la Comisión de Trabajo del Expediente Judicial Electrónico (EJE). Fecha: diciembre 2022.

Otras Actividades de Difusión:

- Parte de los resultados de esta investigación son difundidos en el curso de posgrado "Capacitación en Técnicas de Litigación Digital y Oral para el Fuero Penal". El mismo es organizado por el Colegio de Abogados del Departamento Judicial Morón, tiene una duración de 10 semanas. Modalidad: Virtual. Laura Conti y Sergio García participan como docentes.
- Conti participó como disertante en la Jornada de actualización: *La gestión judicial: Nuevos Desafíos.* Organizada por FUNDEJUS (Fundación de Estudios para la Justicia) y la Asociación de Magistrados y Funcionarios del Departamento Judicial de Morón. Fecha: noviembre de 2022.

- Pérez Villar y Conti fueron disertantes en el evento: *Aspectos prácticos, tecnológicos y normativos del expediente judicial digital de la Provincia de Buenos Aires. Sistemas de gestión en uso. Notificaciones, presentaciones y oficios electrónicos*. Organizada por la Comisión de Derecho Procesal Digital CAM y la Asociación de Magistrados/as y Funcionarios/as de Morón. Fecha: noviembre de 2022.

Formación de Recursos Humanos:

- El alumno Fabio Quintana, finalizó su carrera de Ingeniería en Informática con la aprobación del proyecto final de carrera referido a un sistema para supervisión en tiempo real de la actividad física fin de evitar lesiones por malos movimientos o posturales, recibiendo el título correspondiente en marzo de 2022.
- Los integrantes del proyecto han realizado los siguientes cursos o asistido a jornadas de capacitación durante el transcurso del proyecto:

<u>Ledesma:</u>

- Panel: "Nueva normalidad: ¿Tu equipo legal tiene lo que necesita para ser exitoso?". Modalidad virtual. Organizado por la empresa Thomson Reuters. Duración: 2 hs. Fecha: 7 de julio de 2021.
- Jornada sobre Procesamiento del Lenguaje Natural: "Algoritmos que entienden el lenguaje: Aspectos computacionales y aplicaciones sociales". Modalidad virtual. Organizado por el Departamento de Ciencias Básicas de la Universidad Nacional de Luján. Duración: 3 hs. Fecha: 17 de agosto de 2021.
- Curso: "Introducción a UX". Organizado por: Education IT. Duración: 12 hs. Fecha de finalización: octubre de 2021.
- Curso: "UI: Interfaz de Usuario". Organizado por: Education IT. Duración: 30 hs. Fecha de finalización: noviembre de 2021.
- Jornada: 51 JAIIO. SID 2022 - Simposio Argentino de Informática y Derecho. Organizada por SADIO. Fecha 17 al 27 de octubre 2022.

<u>Bossero</u>:

- Jornada sobre Procesamiento del Lenguaje Natural: "Algoritmos que entienden el lenguaje: Aspectos computacionales y aplicaciones sociales". Modalidad virtual. Organizado por el Departamento de Ciencias Básicas de la Universidad Nacional de Luján. Duración: 3 hs. Fecha: 17 de agosto de 2021.
- Congreso. CACIC 2021. Organizado por Universidad Nacional de Salta. Fecha: 4 al 8 de octubre de 2021.
- Congreso. CONAIISI 2021. Organizado por Universidad Tecnológica Nacional Facultad Regional Mendoza. Fecha: 4 y 5 de noviembre de 2021.
- Workshop: WICC 2022. Organizado por Universidad de Champagnat, Mendoza. Fecha: abril 2022.
- Congreso. CACIC 2022. Organizado por Universidad Nacional de La Rioja. Fecha: 3 al 6 de octubre de 2022.
- Congreso. CONAIISI 2022. Organizado por Facultad Regional Concepción del Uruguay de la Universidad Tecnológica Nacional. Fecha: 4 de noviembre de 2022.
- Curso: "Responsive Web Design & Bootstrap". Organizado por: Education IT. Duración: 12 hs. Fecha de finalización: julio de 2022.

Quintana:

- Curso- "Competencias y Desarrollo del Talentos en la Nube para la Comunidad UNLaM". Organizado por UNLaM. Duración: 30 hs. Fecha de finalización: febrero 2021.
- Curso- "Developing MVC 5 Web Applications". Organizado por Education IT. Duración: XX Fecha de finalización: marzo de 2021.
- Curso- "Vue JS. Organizado por Education IT". Duración: 15 hs. Fecha de finalización: diciembre de 2021.
- Curso- "Node. JS y Mongo DB". Organizado por Education IT. Duración: 18 hs. Fecha de finalización: enero de 2022.

Matteo:

- Webinar- "Phyton". Organizado por: Education IT. Duración: 2 hs. Fecha: 29 de marzo de 2021.
- Curso- "Formación en Oratoria". Organizado por UNLaM, Escuela de Posgrado. Duración: 24 hs. Fecha de finalización: abril 2021.
- Webinar- "Herramientas de evaluación en plataforma MIeL". Organizado por UNLaM. Duración: 1 hora.
- Taller- "Buenas prácticas docentes en espacios virtuales de enseñanza". Organizado por UNLaM, Dirección de Pedagogía Universitaria. Duración: 2 hs. Fecha: 24 de junio de 2021.
- Jornada- "Actualización en Desarrollo Sostenible para Emprendedores". Organizado por UNLaM, Secretaría de Investigaciones del DIIT. Duración: 3 hs. Fecha: 8 de julio de 2021.
- Taller- "Sentidos y sentires de las tutorías en la virtualidad". Organizado por UNLaM, Dirección de Pedagogía Universitaria. Duración: 3 hs. Fecha: 29 de septiembre de 2021.
- Simposio- "Aulas híbridas y bimodalidad en la Educación Superior". Organizado por UNLaM, Dirección de Pedagogía Universitaria. Duración: 2 hs. Fecha: 28 de octubre de 2021.
- Congreso. CACIC 2021. Organizado por Universidad Nacional de Salta. Fecha: 4 al 8 de octubre de 2021.
- Taller- "Estrategias de búsqueda y recuperación de información científica en bases de datos". Organizado por UNLaM, Biblioteca, Leopoldo Marechal. Duración: 3 hs. Fecha: 12 de noviembre de 2021.
- Curso de Capacitación: "Buenas prácticas de consumo y producción de la información científica y académica". Organizado por: Biblioteca Leopoldo Marechal de la UNLaM. Duración: 2 encuentros. Fecha: noviembre 2022.

Moreno:

- Jornada sobre Procesamiento del Lenguaje Natural: "Algoritmos que entienden el lenguaje: Aspectos computacionales y aplicaciones sociales". Modalidad virtual. Organizado por el Departamento de Ciencias Básicas de la Universidad Nacional de Luján. Duración: 3 hs. Fecha: 17 de agosto de 2021.
- Curso: "Programación .NET con C# .NET". Organizado por: Education IT. Duración: 40 hs. Fecha de finalización: mayo de 2022.

Gargano:

- Curso- "Phyton Programming". Organizado por: Education IT. Duración: 18 hs. Fecha de finalización: noviembre de 2021.

- Curso- "Phyton para Análisis de Datos". Organizado por: Education IT. Duración: 18 hs. Fecha de finalización: febrero de 2022.

Saizar:

- Curso: "ReactJS Developer". Organizado por: Education IT. Duración: 36 hs. Fecha de finalización: julio de 2022.
- Curso: "Web API. Net Core". Organizado por: Education IT. Duración: 21 hs. Fecha de finalización: julio de 2022.

Conti:

- Curso AI0101SP: Inteligencia Artificial para todos: Domina los fundamentos. Organizado por IBM mediante edX. Duración 8 hs. Fecha de finalización: febrero de 2022.
- Curso Superior en Derecho. Inteligencia Artificial y Derecho. Organizado por la Fundación General de la Universidad de Salamanca (España), y avalado por la Comunidad Europea. Duración: 120 hs. Fecha de Finalización: noviembre de 2022.

- Además de lo detallado previamente, todos los integrantes del equipo realizaron la capacitación obligatoria en género y violencia de género establecida en la Ley Micaela (conforme a Ley 27.499).

Direcciones y Tutorías de alumnos de grado y posgrado:

- Bossero: Director de Tesis de Maestría del Ing. Mauro Casuccelli. Título: "Estudio comparativo de DBSCAN, KMEANS con redes neuronales en un Sistema de Recuperación deInformación". Maestría en Informática, Escuela de Posgrado, Universidad Nacional de La Matanza. Inicio: agosto 2018. En desarrollo.

Otras Actividades Científicas y Tecnológicas:

- Sposito, Ledesma, Procopio, Conti, García y Perez Villar - Participaron en la organización de una jornada titulada "Experticia. Un camino hacia la Inteligencia Artificial en la justicia", realizada en forma conjunta por la Universidad Nacional de La Matanza y el Colegio de Magistrados y Funcionarios de la Provincia de Buenos Aires. Fue llevada a cabo el 31 de agosto de 2021 a través de la Plataforma Zoom hubo alrededor de 150 conexiones y además se transmitió por YouTube, con más de 500 vistas. El evento fue declarado de interés por la Suprema Corte de Justicia.

H. Cuerpo de anexos:

- Anexo I: Copia de cada uno de los trabajos mencionados en los puntos B, C y D, y certificaciones cuando corresponda.[4]
- Anexo II:
  - FPI-013: Evaluación de alumnos integrantes. (si corresponde)
  - FPI-014: Comprobante de liquidación y rendición de viáticos. (si corresponde)
  - FPI-015: Rendición de gastos del proyecto de investigación acompañado de las hojas foliadas con los comprobantes de gastos.

[4] En caso de libros, podrá presentarse una fotocopia de la primera hoja significativa o su equivalente y el índice.

- FPI-035: Formulario de reasignación de fondos en Presupuesto.

- Nota justificando baja de integrantes del equipo de investigación.

_____

Firma y aclaración
del director del proyecto.

Lugar y fecha: San Justo, 30 de mayo de 2023

- Cargar este formulario junto con los documentos correspondientes **exclusivamente** al Anexo I en SIGEVA UNLaM. Realizar la presentación impresa de los mismos junto con los restantes Anexos en la Secretaría de Investigación de la unidad académica correspondiente**. Límite de entrega: 28 de febrero de 2020.**

ANEXO I

COPIA DE ARTÍCULOS

Revista Digital del Departamento de Ingeniería e Investigaciones Tecnológicas de la Universidad Nacional de la Matanza
ISSN: 2525-1333
Vol.: 7 - Nro. 2 (DICIEMBRE 2022)

Artículo original

Determinación del umbral inferior de coincidencia aplicando medidas de edición a términos jurídicos

Determination of the lower similarity threshold applying measures of edit distance to legal terms

Lorena Matteo [1], *Viviana Ledesma* [2], *Osvaldo Spósitto* [3]

[1] Universidad Nacional de La Matanza
lmatteo@unlam.edu.ar

[2] Universidad Nacional de La Matanza
vledesma@unlam.edu.ar

[3] Universidad Nacional de La Matanza
spositto@unlam.edu.ar

Resumen:

Aplicar técnicas que ayuden a reducir el espacio de búsqueda en tareas de consultas a corpus jurídicos documentales es sumamente importante debido al volumen y diversidad de datos involucrados. Utilizando medidas de similitud léxica, en particular, aquellas basadas en cadenas de caracteres, es posible encontrar el umbral que determine el límite inferior aceptable del porcentaje de coincidencia de los términos que representan el mismo concepto. De este modo se minimiza la tarea manual de los expertos de dominio, ayudándolos a focalizarse en la revisión/validación de la similitud de aquellos términos que estén dentro de ese umbral de coincidencia. Seleccionando el término más representativo de cada concepto es posible reducir la matriz término-documento, punto de entrada para la búsqueda de información dentro del corpus.

ISSN: 2525-1333
Vol.: 7 - Nro. 2 (DICIEMBRE 2022)

En este artículo se explica el procedimiento para encontrar el umbral de coincidencia que surge al aplicar medidas de similitud léxica a ciertos grupos de términos que representan distintos escenarios jurídicos. Estas medidas son las distancias de edición de Hamming y de Levenshtein.

Los resultados muestran que el umbral puede variar según cada escenario o medida, ayudando a los expertos a centrarse en el análisis de aquellos términos cuyo porcentaje de similitud esté dentro del umbral propuesto.

Abstract:

Applying techniques that help reduce search time in query tasks to documentary legal corpus is of great importance due to the volume and diversity of data involved. Using measures of lexical similarity, based on character strings, it is possible to find the threshold that determines the acceptable lower limit of the coincidence percentage of terms that represent the same concept. In this way, the manual task of domain experts is minimized, helping to focus on the review/validation of the similarity of those terms that are within that matching threshold. By selecting the most representative term for each concept in question, it is possible to reduce the term-document matrix, the entry point for searching for information within the corpus.

This article explains the procedure to find the coincidence threshold that arises when applying lexical similarity measures to certain groups of terms that represent different legal scenarios. These measures are the Hamming and Levenshtein edit distances.

The results show that the threshold can vary according to each scenario/measurement, helping experts to focus on the analysis of terms whose percentage of similarity is within the proposed threshold.

Palabras Clave: *Medidas de Similitud Léxica; Umbral de Similitud; Sistema de Recuperación de Información; Hamming; Levenshtein*

Key Words: *Lexical Similarity Measures; Similarity Threshold; Information Retrieval System; Hamming; Levenshtein*

Colaboradores: *Julio Bossero, Edgardo Moreno*

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

I. CONTEXTO

Este artículo se enmarca en una línea de investigación, relacionada al estudio de los Sistemas de Recuperación de Información (SRI) realizada por investigadores del Departamento de Ingeniería e Investigaciones Tecnológicas y del Departamento de Derecho y Ciencia Política de la Universidad Nacional de La Matanza. Particularmente se asocia al proyecto PROINCE, código C241, "*Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado*", con vigencia 2021-2022.

II. INTRODUCCIÓN

En el dominio judicial, la jurisprudencia es un factor importante como fuente de derecho; porque sus conclusiones crean una pauta para la aplicación de la ley ante situaciones jurídicas similares. Cada año el poder judicial argentino produce una gran cantidad de decisiones que se guardan en diversas formas, como ser dictámenes o expedientes, haciendo que esta fuente documental sea cada vez más voluminosa, lo que impulsa a los profesionales de la justicia a dedicar más tiempo a la búsqueda de documentos relevantes. Esto conduce a la aplicación de técnicas sofisticadas para reducir el tiempo de búsqueda y mejorar la pertinencia de los documentos recuperados.

En tal contexto, como se mencionó previamente, este grupo de investigación se encuentra trabajando en la especialización de un SRI para su utilización en un contexto jurídico. El principal objetivo es que dicho sistema permita, a partir de una consulta, recuperar documentos con características similares y útiles para la resolución de un problema legal. A su vez, se pretende diseñar y crear un corpus de referencia jurídica.

Como es sabido, es de suma importancia contar con la participación de los expertos de dominio para ir validando los términos que forman parte del corpus, muchas veces efectuando controles manuales, lo cual implica un esfuerzo considerable.

A fin de reducir ese costo de intervención manual y, para mejorar la performance en la búsqueda exhaustiva de patrones realizada inicialmente mediante el uso de expresiones regulares [1], es que surge la necesidad de aplicar técnicas complementarias que ayuden a reducir la dimensionalidad o cantidad de términos de la matriz término-documento. Dicha matriz es el punto de entrada en la búsqueda de información dentro un corpus, según se explicará más adelante.

Por tanto, si bien el procedimiento y acciones descriptos en el presente artículo surgen en principio con el objetivo de reducir los términos de la matriz de búsqueda de documentos jurídicos, esto puede aplicarse para encontrar un umbral de similitud optimo entre cualquier conjunto de términos, sea de la índole que fuera. Esto ayudará a los expertos de dominio a centrarse en el análisis manual de aquellos términos cuyo porcentaje de coincidencia se encuentre dentro del umbral sugerido.

III. SISTEMAS DE RECUPERACIÓN DE INFORMACIÓN

Un SRI, puede describirse como un conjunto de ítems de información o corpus de documentos, un conjunto de peticiones y un mecanismo que determine qué ítems satisfacen las peticiones de los usuarios. En otras palabras,

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

devuelve una lista ordenada o rankeada de documentos supuestamente relevantes para la consulta [2].

Se han ideado diferentes modelos basados en distintos paradigmas para la representación de un SRI, así como para calcular el grado de similitud entre los elementos de información para responder determinada consulta [3] [4] [5].

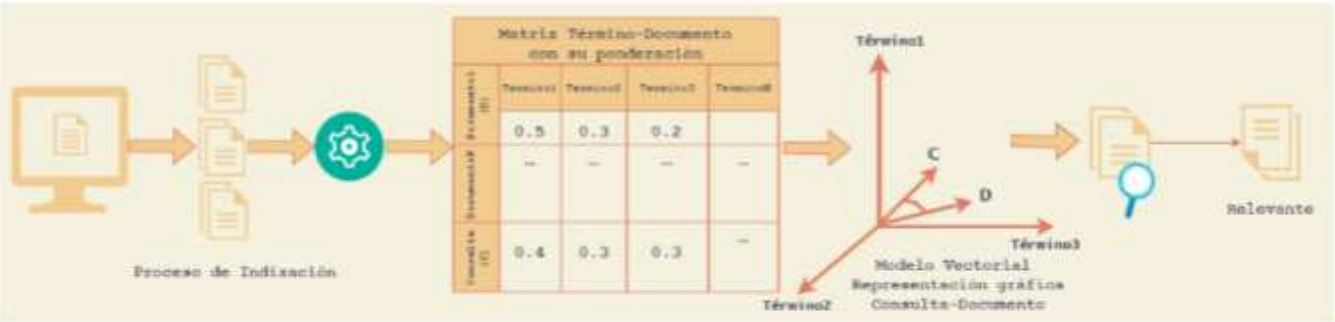


Figura 1. Proceso de búsqueda en un SRI. Fuente: Elaboración propia.

Hay tres modelos, que se consideran clásicos y son los más utilizados:

- *Booleano*: basado en la teoría de conjuntos y en el algebra de Boole. Se crea un conjunto con los elementos de la consulta y otro con los documentos, posteriormente se mide la correspondencia.
- *Vectorial*: se apoya en la idea de la importancia de un término con respecto a un documento, así como que los documentos y las consultas se pueden representar como un vector en un espacio de alta dimensionalidad. De esta manera, la consulta y los términos del documento se representan mediante dos vectores, midiéndose el grado de similitud entre ambos.
- *Probabilístico:* se calcula la probabilidad en que el documento responde a la consulta. Frecuentemente se usa retroalimentación, mediante interacción con el usuario para que indique qué documentos son más relevantes, para así reformular la consulta.

El trabajo de investigación en curso se enmarca en un modelo vectorial. Como se grafica en la Figura 1, la colección o corpus documental se representa en una matriz de término-documento. En la intersección de un término y un documento se almacena un valor numérico para denotar la importancia de tal término en el documento. Así, cada documento puede ser visto como un vector que pertenece a un espacio *n-dimensional*, donde n es la cantidad de términos que componen el vocabulario del corpus. En teoría, los documentos que contengan términos similares estarán cercanos entre sí sobre tal espacio. Una consulta se considera un documento más y se la mapea sobre el espacio de documentos. Entonces, a partir de una consulta dada es posible devolver una lista de documentos ordenados por distancia con los más relevantes primero.

En cualquier dominio de conocimiento, aquellos términos con significado pueden servir como descriptores para una representación lógica del contenido de documentos, así

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

como para las consultas en el proceso de recuperación de información [6]. Por lo tanto, una fase muy importante en un SRI es la de preparación de los documentos, esto implica que, en la fase de entrada, se deba realizar la indización y organización de la información. Según Tolosa y Bordignon, en [7] afirman que dicho proceso se puede dividir en las siguientes etapas:

- Análisis lexicográfico, se extraen las palabras y se normalizan.
- Reducción (Tokenización) de palabras vacias o de alta frecuencia.
- Lematización, se reducen palabras morfológicamente parecidas a una forma base o raiz, con la finalidad de aumentar la eficiencia de un SRI.
- Selección de los términos a indexar. Se extraen aquellas palabras simples o compuestas que mejor representan el contenido de los documentos.
- Asignación de pesos o ponderación de los términos que componen los índices de cada documento.

El trabajo que se está presentando en el presente artículo se circunscribe al proceso antes mencionado. El SRI desarrollado por este grupo de investigación, adopta inicialmente una de las representaciones más extendidas, sobre todo por su simpleza, la matriz término-documento, también llamada 'bolsa de palabras'. Es decir, se forma un vector con la frecuencia de los términos del texto, con lo cual, los documentos se caracterizan por las palabras que contienen [8].

IV. REDUCCIÓN DE DIMENSIONALIDAD DE LA MATRIZ TÉRMINO-DOCUMENTO

Las matrices conseguidas con la bolsa de palabras tienen una gran cantidad de variables o dimensiones, por no estar normalizadas, lo cual es poco útil para trabajar. Por ello, se busca una reducción de dimensionalidad, esto es llevar a una mínima cantidad posible, el número total de dimensiones que existen en el modelo del espacio vectorial.

A partir de dicha representación lógica del corpus, mediante el proceso de indización se lleva a cabo la construcción de estructuras de datos o índices a fin de brindar posteriormente soporte para la recuperación de los documentos.

Con lo anterior presente, este equipo ha propuesto, como parte de esta investigación, un algoritmo para la búsqueda y reemplazo de Entidades Nombradas (EN) utilizando Expresiones Regulares (ER). Una ER es una secuencia de caracteres que forma un patrón de búsqueda. Una EN, según [9], "*es una palabra o secuencias de palabras que se identifican como nombre de persona, organización, lugar, fecha, tiempo, porcentaje o cantidad...*". Mayor detalle de esta propuesta se incluyó en [1], donde se analizó implementar en el proceso de indización de términos de un corpus jurídico, la identificación de fechas y de referencias a EN, tales como Expediente Nº, Resolución Nº, Artículo Nº de la Ley XXX, que remiten a la norma jurídica vigente y son ampliamente utilizadas en distintos documentos judiciales.

En dicho trabajo se concluye que la aplicación de ER para encontrar EN tiene la como ventaja que:

- una vez hallada la expresión correcta, las entidades que durante la búsqueda exhaustiva

coincidan exactamente con dicho patrón serán todas las existentes en el corpus. Esto tiene mayor importancia dado que los textos legales son muy estructurados y las entidades aparecen con cierta regularidad, por otra parte,

- son fáciles de implementar ya que no necesitan más que codificar la expresión del patrón en si, y no requieren, por ejemplo, del entrenamiento de un modelo para su reconocimiento.

Como desventaja, se sabe que estas se limitan a encontrar los patrones predefinidos, por lo cual, no es posible encontrar otra EN que no coincida con alguna de las ER existentes.

Por esta razón y considerando que el SRI debe procesar lenguaje natural, es que para ir un paso más adelante, se encaró la tarea de comparar los términos entre si, resultando de interés detectar no sólo las coincidencias exactas entre dos términos, sino también disponer de una medida de aproximación o similitud entre estos para los casos en que la coincidencia no sea exacta. Se puso en foco la detección de términos jurídicos similares, los cuales surgen como resultado del proceso de indización y organización de la información del SRI, de este modo se pretende reducir el esfuerzo de seleccionar manualmente los términos a indexar.

Las palabras pueden ser similares léxica o semánticamente. La similitud léxica toma en cuenta si las palabras tienen secuencias de caracteres semejantes [10]. Por otro lado, las palabras tienen similitud semántica si significan lo mismo en un contexto dado, aunque léxicamente sean distintas.

Las funciones de similitud léxica han sido investigadas por décadas, existen diversos métodos o propuestas para la resolución del cálculo de la similitud de este tipo, cada una tiene sus peculiaridades según la aplicación que se le deba dar [3] [11]. Según Elmagarmid y otros [12], las distintas propuestas podrían dividirse en dos grupos: las basadas en cadenas de caracteres (distancia de edición, Brecha Afin, Smith-Waterman, Hamming, Levenshtein y Jaro, entre otras) y las basadas en tokens o secuencias de palabras (por ejemplo, Similitud de Monge-Elkan y Similitud coseno TF-IDF).

El análisis que se está presentando en este artículo se enfoca en la similitud léxica basada en cadenas de caracteres, dentro de este grupo, en la distancia de edición. Esta se define como la cantidad mínima de cambios requeridos para transformar la cadena origen en la cadena destino, en donde las operaciones permitidas se eligen de un conjunto fijo como ser la eliminación, inserción y sustitución. Como se adelantó, en este trabajo se presentan parcialmente los resultados obtenidos al comparar dos de las métricas más utilizadas en esta categoría, la distancia de Hamming (HAM) y la distancia de Levenshtein (LEV), con el objetivo de reducir la dimensionalidad de la matriz término-documento respecto de aquellos términos coincidentes. Para lograrlo, se busca encontrar un umbral de similitud aceptable que permita asumir que dos términos son representaciones de la misma EN. Ese umbral surge de comparar el porcentaje de similitud, basado en dos métricas de Precisión y Recall, ampliamente utilizadas en este tipo de ensayos como se efectuó en [16]. Estas ayudan a determinar la efectividad de las técnicas de detección de similitud de cadenas. De este modo, los expertos de dominio pueden centrarse en el análisis de aquellos términos cuyo porcentaje de similitud

esté dentro del umbral propuesto, a mayor sea su valor, mayor similitud entre los términos

IV. DISEÑO DEL EXPERIMENTO PARA LA DETERMINACIÓN DEL UMBRAL

En esta sección se explica el método aplicado para encontrar el umbral de similitud de cada grupo de términos jurídicos, creados para tal fin, en base a la efectividad resultante de las medidas de edición de caracteres.

- *Paso 1: creación de grupos de términos para representar distintos escenarios jurídicos.*

Para llevar adelante este trabajo se ha partido de una lista de 11.155 términos, es decir EN, resultantes del proceso de indización y organización de la información del SRI. La lista original contiene 3 campos: clave, término y ocurrencia en el corpus.

Con el objetivo de reducir esa lista de términos se ha recurrido a las técnicas de detección de similitud de cadenas. A modo de ensayo, se utilizó un procedimiento basado en experimentos, para ello, tal como se refleja en la Tabla 1, se armaron 5 grupos de EN significativas.

Como se mencionó anteriormente, las funciones de similitud elegidas para abordar este trabajo fueron HAM y LEV. La comparación se realizó mediante las métricas de evaluación de efectividad: precisión y recall, buscando determinar la eficacia de las funciones ante cada escenario, representado por cada grupo de entidades nombradas, Grupo EN_x, donde x identifica el caso de estudio.

Tabla 1.
Composición de experimentos por grupos de EN

Caso Estudio	Grupo EN	Cantidad Términos	Concepto
1	Grupo EN1	12	Relación con EN "Legal"
2	Grupo EN2	8	Relación con EN "Oficial"
3	Grupo EN3	16	Relación con EN "Expediente"
4	Grupo EN4	9	Relación con EN "Mediación"
5	Grupo EN5	15	Relación con EN "Fechas y Otras"
	Total EN	60	

- *Paso 2: clasificación manual de similitud entre los términos de cada Grupo de EN_x*

Para aplicar las métricas de evaluación de efectividad mencionadas es necesario que los expertos de dominio clasifiquen previamente las coincidencias reales entre cada par de términos incluidos en cada uno de los grupos de EN. Para ello se armaron matrices, donde las EN de cada grupo se colocaron en las filas y se repitieron en las columnas. Un ejemplo de ello se puede visualizar en la Figura 2, donde se muestra la clasificación de similitud para el grupo EN_2. En la intersección de cada término el experto debió realizar una clasificación manual de las coincidencias en reales/verdaderas y falsas, asignando el valor 1, cuando los considere similares o 0 en caso contrario.

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

EXPEREMIENTO CLASIFICACION MANUAL

Terminos GrupoEN2	boletinoficial	filosoficoreligi	ofici	oficial	oficializ	oficialy	oficin	suboficial
boletinoficial	1	0	0	0	0	0	0	0
filosoficoreligi	0	1	0	0	0	0	0	0
ofici	0	0	1	1	1	1	1	0
oficial	0	0	1	1	1	1	0	0
oficializ	0	0	1	1	1	1	0	0
oficialy	0	0	1	1	1	1	0	0
oficin	0	0	1	0	0	0	1	0
suboficial	0	0	0	0	0	0	0	1

Figura 2. Clasificación Manual de Similitud entre los términos del Grupo EN_2. Fuente: Elaboración propia.

Esta clasificación fue útil para comparar el resultado conseguido más tarde con la aplicación de las funciones de HAM y LEV a cada grupo, siendo dicho resultado el porcentaje de coincidencia de cada término de la matriz de similitud. De esta forma es posible evaluar la efectividad de los porcentajes de similitud, encontrando el límite inferior del umbral de coincidencia. El beneficio de esto radica en que cuando los expertos deban analizar el corpus completo, puedan enfocarse en el análisis de términos cuyo porcentaje de similitud esté dentro del umbral propuesto, reduciendo de este modo su carga de trabajo.

- *Paso 3: Cálculo de la distancia de HAM*

Esta métrica se basa en [13], es igual a la cantidad de posiciones en las que difieren ambas cadenas, y sólo permite la sustitución. Se obtiene haciendo un conteo del número de posiciones en las que los caracteres de las cadenas comparadas difieren, siendo 0 el valor resultante cuando hay coincidencia total entre las cadenas y, distinto de 0 en caso contrario. Es útil para comparar dos cadenas de caracteres de igual longitud. Es una de las métricas más simples en la que se considera el orden de los elementos.

Con lo anterior presente, el estudio realizado requirió que en un inicio las EN a comparar se ordenaran alfabéticamente. Además, se agregaron espacios en blanco a aquellas EN de la cadena de menor longitud para equiparar la cantidad de caracteres, respetando de este modo la restricción de HAM.

Se implementó una función para el cálculo, los resultados obtenidos a partir de la misma se expresaron en porcentajes de coincidencia, a modo de ejemplo, en la Figura 3 se puede observar los valores resultantes correspondientes al grupo EN_2.

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

Terminos GrupoEN2	boletinoficial	filosoficoreligi	ofici	oficial	oficializ	oficialy	oficin	suboficial
boletinoficial	100,0	6,0	0,0	0,0	0,0	0,0	0,0	7,0
filosoficoreligi	6,0	100,0	0,0	0,0	6,0	0,0	0,0	12,0
ofici	0,0	0,0	100,0	71,0	56,0	62,0	83,0	0,0
oficial	0,0	0,0	71,0	100,0	78,0	88,0	71,0	0,0
oficializ	0,0	6,0	56,0	78,0	100,0	78,0	56,0	10,0
oficialy	0,0	0,0	62,0	88,0	78,0	100,0	62,0	0,0
oficin	0,0	0,0	83,0	71,0	56,0	62,0	100,0	0,0
suboficial	7,0	12,0	0,0	0,0	10,0	0,0	0,0	100,0

Figura 3. Porcentaje de Similitud entre términos del Grupo EN_2 usando HAM. Fuente: Elaboración propia.

Para facilitar la visualización durante el análisis de estos ensayos se utilizaron colores tipo semáforo para diferenciar los porcentajes de coincidencia: siendo la gama de verdes, según su intensidad, los más cercanos al 100%.

- *Paso 4: Cálculo de la distancia de LEV*

Esta distancia o indice también pertenece a las distancias de edición, siendo el resultado de este algoritmo dinámico la cantidad mínima de operaciones que se requiere para convertir un término en otro; entendiéndose por operaciones de edición a la inserción, eliminación o sustitución de caracteres dentro de esa EN, según se explica en [14]. Mientras mayor sea la distancia de LEV, mayor será la diferencia entre los dos términos; por ende, y al igual que en HAM, una distancia de valor igual a 0 indica que los dos términos son idénticos. Para el estudio, del mismo modo que se hizo con HAM, se implementó la función para el cálculo de LEV, en la Figura 4 se pueden observar los porcentajes resultantes para el grupo EN_2.

Tal lo mencionado en [15], esta técnica se destaca por su capacidad de detección de errores tipográficos típicos, en dicho artículo se encuentran categorizados como situaciones problemáticas, a saber: errores ortográficos y tipográficos, abreviaturas: truncamiento de uno o más términos, términos faltantes, eliminación de uno o más términos, prefijos/sufijos sin valor semántico, términos en desorden, espacios en blanco. Al trabajar con un corpus asociado a un contexto jurídico estas situaciones podrían ser corrientes, por lo que detectar la similitud entre términos con estas características es parte del objetivo de este experimento, ya que al detectarlos el experto puede decidir cuál de esos términos mejor representa a la EN en cuestión.

Revista Digital del Departamento de Ingeniería e Investigaciones Tecnológicas de la Universidad Nacional de la Matanza
ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

Terminos GrupoEN2	boletinoficial	filosoficoreligi	ofici	oficial	oficializ	oficialy	oficin	suboficial
boletinoficial	100	25	36	50	36	43	36	50
filosoficoreligi	25	100	31	31	38	31	31	31
ofici	36	31	100	71	56	62	83	50
oficial	50	31	71	100	78	88	71	70
oficializ	36	38	56	78	100	78	56	50
oficialy	43	31	62	88	78	100	62	60
oficin	36	31	83	71	56	62	100	50
suboficial	50	31	50	70	50	60	50	100

Figura 4. Porcentaje de Similitud entre términos del Grupo EN_2 usando LEV. Fuente: Elaboración propia.

- *Paso 5: Obtención de medidas de Precisión y Recall*

A fin de comparar la eficacia en la detección de coincidencias utilizando las dos funciones comparadas, se aplicaron las métricas, Precisión y Recall, que permiten encontrar el límite inferior del umbral de coincidencia ya mencionado.

Acá entra en juego la clasificación manual efectuada por el experto de dominio mencionada en el Paso 2 de este apartado.

Precisión trata de responder la pregunta: ¿Qué proporción de los términos identificados como coincidentes son realmente correctos?

Es la relación entre el número de términos coincidentes identificados correctamente y el número total de términos coincidentes que ha identificado la función (de HAM o de LEV).

$$Precisión = \frac{coincidentes\ identificados\ correctamente}{número\ coincidentes\ identificados}$$

Recall, por su parte, trata de responder la pregunta: ¿Qué proporción de los términos coincidentes reales se identifica correctamente?

Es la relación entre el número de términos coincidentes identificados correctamente y el número de coincidentes que realmente hay en el grupo EN_x.

$$Recall = \frac{coincidentes\ identificados\ correctamente}{número\ coincidentes\ reales}$$

Estas medidas se calculan para cada rango de porcentajes, etiquetándolos como se ve en las Figuras 5 y 6.

PrecisiónUmbralMen60	PrecisiónUmbralMay60	PrecisiónUmbralMay70	PrecisiónUmbralMay80	PrecisiónUmbralMay90	PrecisiónUmbral100

Figura 5.Precisión para los Rangos de Umbrales de 0% a 100%.

Fuente: Elaboración propia.

RecallUmbralMen60	RecallUmbralMay60	RecallUmbralMay70	RecallUmbralMay80	RecallUmbralMay90	RecallUmbral100

Figura 6. Recall para los Rangos de Umbrales de 0% a 100%.
Fuente: Elaboración propia.

De acuerdo con los valores obtenidos, como se puede ver en la Figura 7, se descarta la columna del umbral menores al 60%, columnas *"Precisión y Recall UmbralMen60"* dado que se estaría generalizando demasiado los términos del corpus, asumiendo esas coincidencias como válidas cuando en realidad no lo son. Detenerse en ello no sería útil para los expertos dado que revisarían términos sin relación alguna y, por ende, términos que no podrían eliminarse del listado. Cabe recordar que el objetivo principal de este trabajo es facilitar las tareas reducción de la dimensionalidad de la matriz término-documento. Por otra parte, luego de los resultados obtenidos, los cuales

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

seguirán en estudio, también se descartan las columnas *"Precisión y Recall UmbralMay90"* dado que no se encontraron términos en ese rango de similitud. Finalmente, se descartan los resultados del umbral igual al 100%, columna "*Precisión y Recall Umbral100"*, ya que en dicha columna se ubican todos los términos que pertenecen a la diagonal principal de la matriz de coincidencias, es decir el cruce de cada término consigo mismo. Por tanto, los umbrales a analizar serán los comprendidos entre 60 y 99%, siempre observando los términos en orden alfabético.

- *Paso 6: Consolidación de Resultados*

A fin de visualizar los resultados de manera más clara y concisa, tal como se puede ver en las Figuras 8 y 9, se confeccionó un tablero que consolida los resultados de las métricas de efectividad resultantes para los umbrales seleccionados. Se incluye los promedios para cada una de las distancias de HAM y LEV por grupo EN_x. De esta forma es posible analizar a simple vista aquellos grupos EN_x cuyos porcentajes de similitud sean los mayores dentro de los umbrales seleccionados, ya que esos grupos contendrán términos relevantes para las tareas de revisión y reducción de las EN

GrupoEN	MetodoEdicion	Termino	CantSimilares Reales	PrecisionUmbralMay60	PrecisionUmbralMay60	PrecisionUmbralMay70	PrecisionUmbralMay80	PrecisionUmbralMay90	PrecisionUmbral100	RecallUmbralMay60	RecallUmbralMay60	RecallUmbralMay70	RecallUmbralMay80	RecallUmbralMay90	RecallUmbral100
GrupoEN4	Hamming	mediar	6	25%	100%	0%	100%	0%	100%	17%	33%	0%	33%	0%	17
GrupoEN4	Hamming	mediat	3	0%	0%	0%	100%	0%	100%	0%	0%	0%	67%	0%	33
GrupoEN5	Hamming	24/9/1991	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	18/10/1993	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	18/10/1993	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	22/11/2001	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	27/2/2006	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	27/2/2006	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	aÃ±o	2	0%	0%	0%	100%	0%	100%	0%	0%	0%	50%	0%	50
GrupoEN5	Hamming	aÃ±os	2	0%	0%	0%	100%	0%	100%	0%	0%	0%	50%	0%	50
GrupoEN5	Hamming	abiert	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	aoon	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	aborda	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	acced	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	acces	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	acept	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN5	Hamming	acers	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN1	Levenshtein	alegar	2	0%	0%	0%	50%	0%	100%	0%	0%	0%	100%	0%	100
GrupoEN1	Levenshtein	alegat	2	0%	0%	0%	100%	0%	100%	0%	0%	0%	50%	0%	50
GrupoEN1	Levenshtein	delegacion	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	50%	0%	50
GrupoEN1	Levenshtein	llegal	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	50%	0%	50
GrupoEN1	Levenshtein	legacion	3	0%	100%	0%	0%	0%	100%	0%	33%	0%	33%	0%	33
GrupoEN1	Levenshtein	legal	9	0%	0%	100%	100%	0%	100%	0%	0%	33%	33%	0%	33
GrupoEN1	Levenshtein	legajo	2	0%	0%	100%	0%	0%	100%	0%	0%	50%	0%	0%	50
GrupoEN1	Levenshtein	legal	4	0%	0%	100%	200%	0%	100%	0%	0%	25%	50%	0%	25
GrupoEN1	Levenshtein	legales	3	0%	100%	100%	0%	0%	100%	0%	33%	33%	0%	0%	33
GrupoEN1	Levenshtein	legatari	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN1	Levenshtein	llegar	1	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN1	Levenshtein	supralegal	2	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%	100
GrupoEN2	Levenshtein	boletinoficial	1	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0

Figura 7. Precisión y Recall para los Rangos de Umbrales de 0% a 100%. Fuente: Elaboración propia.

Revista Digital del Departamento de
Ingeniería e Investigaciones
Tecnológicas de la Universidad
Nacional de la Matanza
ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

Precisión y Recall Hamming por Umbrales (Entre 60 y 89%)

Etiquetas de fila	Promedio de PrecisiónUmbral May60	Promedio de PrecisiónUmbral May70	Promedio de PrecisiónUmbral May80	Promedio de RecallUmbralMay 60	Promedio de RecallUmbralMay 70	Promedio de RecallUmbralMay 80
GrupoEN1	16,67%	33,33%	29,17%	5,56%	11,81%	17,36%
GrupoEN2	18,75%	45,83%	50,00%	5,63%	18,13%	15,00%
GrupoEN3	39,27%	9,38%	9,38%	26,56%	3,65%	3,65%
GrupoEN4	50,00%	22,22%	55,56%	22,04%	7,41%	24,07%
GrupoEN5	0,00%	0,00%	13,33%	0,00%	0,00%	6,67%
Total general	**23,81%**	**18,61%**	**26,67%**	**12,25%**	**6,86%**	**11,72%**

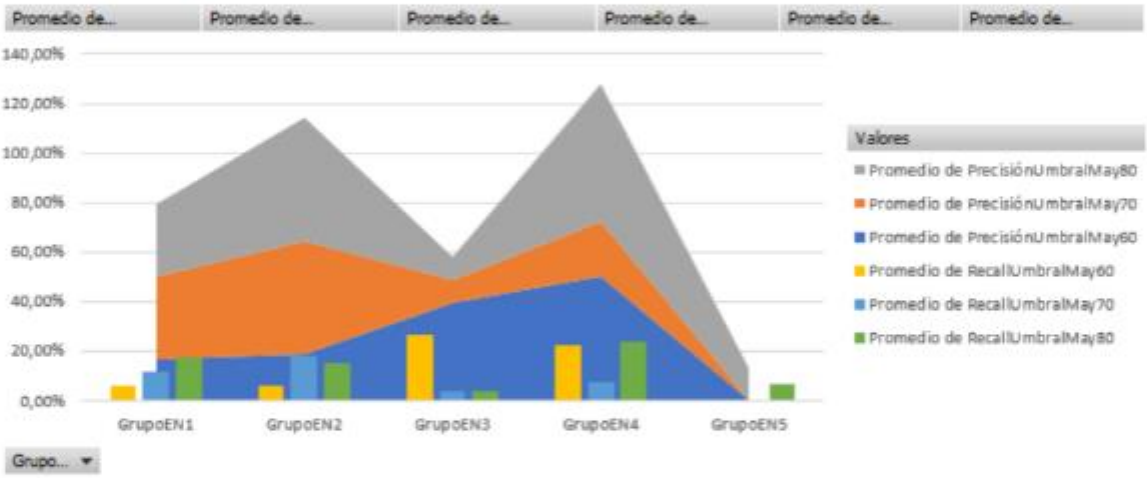


Figura 8. Tablero de control - Umbrales HAM. Fuente: Elaboración propia.

Revista Digital del Departamento de
Ingeniería e Investigaciones
Tecnológicas de la Universidad
Nacional de la Matanza
ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

Precisión y Recall Levenshtein por Umbrales (Entre 60 y 89%)

Etiquetas de fila	Promedio de PrecisiónUmbralMay60	Promedio de PrecisiónUmbralMay70	Promedio de PrecisiónUmbralMay80	Promedio de RecallUmbralMay60	Promedio de RecallUmbralMay70	Promedio de RecallUmbralMay80
GrupoEN1	16,67%	33,33%	37,50%	5,56%	11,81%	30,56%
GrupoEN2	20,83%	41,67%	0,00%	15,63%	15,63%	3,13%
GrupoEN3	18,45%	6,25%	6,25%	23,44%	4,17%	3,13%
GrupoEN4	36,11%	0,00%	11,11%	20,19%	5,56%	7,41%
GrupoEN5	6,67%	0,00%	0,00%	6,67%	0,00%	0,00%
Total general	**18,12%**	**13,89%**	**10,83%**	**14,14%**	**6,39%**	**8,47%**

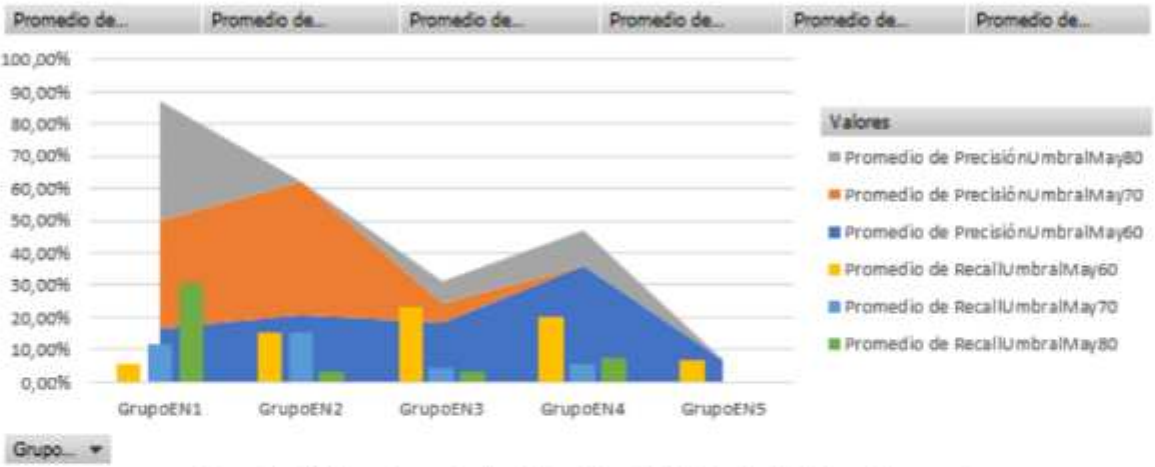


Figura 9. Tablero de control - Umbrales LEV. Fuente: Elaboración propia.

V. ANÁLISIS DE LOS RESULTADOS

Finalmente, para el análisis de los resultados del estudio realizado, se unificaron los resultados obtenidos en un tablero definitivo EN_x (ver Figura 10), lo cual permite comparar los promedios para ambas distancias de edición. Como se puede observar, la precisión de las distancias de HAM y LEV en el grupo EN_1, para el umbral mayor a 70%, es coincidente en un 33% y en aproximadamente en un 12% para el recall. En tanto, la precisión del grupo EN_2 para dicho umbral es de las mayores arrojadas por las medidas de edición. Además, el recall para ese grupo de medidas también es de los más altos. Esto denota la relevancia de que los expertos revisen la similitud de los términos que componen tal grupo. Esto es de suma importancia recordando que esa medida de efectividad indica que proporción de los términos coincidentes reales se identificó correctamente.

Gráficamente, el análisis de resultados es mucho más visible, siendo el umbral del 80%, tanto para la precisión como para recall, el porcentaje de similitud más relevante, en especial para los grupos EN_1, EN_2 y EN_4. Basándose en estos resultados, los expertos de dominio podrán

Revista Digital del Departamento de Ingeniería e Investigaciones Tecnológicas de la Universidad Nacional de la Matanza
ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

focalizarse en las tareas de revisión y reducción de EN de dichos grupos en dicho umbral.

VI. CONCLUSIONES

A través del presente artículo se exhibieron los resultados obtenidos de un estudio llevado a cabo para la facilitar el proceso de detección de términos jurídicos similares.

La motivación que llevó a realizar esta actividad deriva de la importancia de acotar la matriz de término-documento, punto de entrada para el proceso de indexación, necesario para la búsqueda en el contexto de un SRI. El enfoque se puso en la búsqueda del mejor umbral de coincidencia que surge de aplicar medidas de similitud léxica a los términos resultantes del proceso de indización y organización del corpus con el que se está trabajando en la actualidad.

Precisión y Recall Hamming vs Levenshtein por Umbrales 70 y 80%

Etiquetas de fi	Promedio de PrecisiónUmbralMay70	Promedio de PrecisiónUmbralMay80	Promedio de RecallUmbralMay70	Promedio de RecallUmbralMay80
GrupoEN1	**33,33%**	**33,33%**	**11,81%**	**23,96%**
Hamming	33,33%	29,17%	11,81%	17,36%
Levenshtein	33,33%	37,50%	11,81%	30,56%
GrupoEN2	**43,75%**	**25,00%**	**16,88%**	**9,06%**
Hamming	45,83%	50,00%	18,13%	15,00%
Levenshtein	41,67%	0,00%	15,63%	3,13%
GrupoEN3	**7,81%**	**7,81%**	**3,91%**	**3,39%**
Hamming	9,38%	9,38%	3,65%	3,65%
Levenshtein	6,25%	6,25%	4,17%	3,13%
GrupoEN4	**11,11%**	**33,33%**	**6,48%**	**15,74%**
Hamming	22,22%	55,56%	7,41%	24,07%
Levenshtein	0,00%	11,11%	5,56%	7,41%
GrupoEN5	**0,00%**	**6,67%**	**0,00%**	**3,33%**
Hamming	0,00%	13,33%	0,00%	6,67%
Levenshtein	0,00%	0,00%	0,00%	0,00%
Total general	**16,25%**	**18,75%**	**6,63%**	**10,10%**

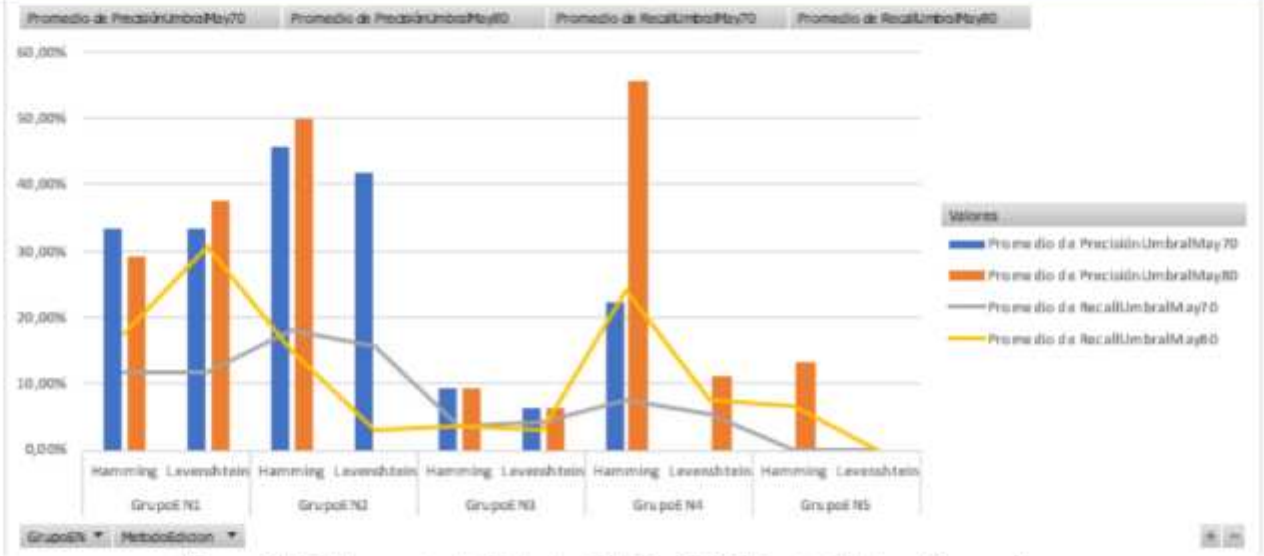


Figura 10. Tablero control - Umbrales HAM y LEV. Fuente: Elaboración propia.

ISSN: 2525-1333. Vol.: 7 - Nro. 2 (DICIEMBRE-2022)

Aunque este procedimiento no exime de la necesidad de contar con el experto humano, puede ser de gran ayuda para minimizar el esfuerzo implicado en su trabajo, debido a que permite acotar el volumen de términos en los que debe centrarse para elegir aquel que mejor represente a la EN.

Se ha observado que los resultados son dependientes de los términos incluidos en cada grupo de EN, y deben analizarse dentro del contexto de cada uno de los escenarios jurídicos creados.

Vale la pena mencionar que no es fácil proporcionar una solución automática, ya que se deben aplicar y adaptar varias técnicas de similitud para adecuarse a los datos concretos de que se disponen. En cuanto a este ensayo, en particular, se concluye que el límite inferior del umbral de coincidencia más relevante es el 80%. De todos modos, y como ya se ha mencionado, ha de tenerse en cuenta qué medida de edición es más fiable en cada escenario a saber: HAM debe considerarse para situaciones donde importa el orden de los caracteres de los términos en cuestión, por ejemplo, fechas, números de leyes, de expedientes, y así por el estilo. Si bien estos también pueden encontrarse fácilmente usando ER, las distancias de edición son más flexibles. En cuanto a LEV, es importante destacar que es más útil para detectar las situaciones problemáticas ya mencionadas, en dichos casos es conveniente mirar los términos del umbral con mejor precisión y recall para esa medida de edición.

VII. TRABAJOS FUTUROS

Para avanzar en esta investigación, a futuro se espera trabajar con un corpus de expedientes jurídicos de mayor volumen, así también, ampliar las medidas de similitud utilizadas.

A su vez, es necesario un análisis más exhaustivo sobre ciertos resultados llamativos dentro del tablero de control consolidado, como ser los promedios de 0% del grupo EN_5 en el umbral del 70%. Así también, será objeto de estudio el análisis de la precisión y recall para el umbralMay90, y la causa por la que no se encontraron términos en ese rango de similitud; probablemente sea necesario incluir mayor cantidad de escenarios jurídicos y por ende EN dentro de estos grupos de estudio.

Por otra parte, es necesario involucrar a mayor cantidad de expertos del dominio, para la validación de los resultados obtenidos, de cara a lograr la automatización del proceso de búsqueda del umbral de coincidencia del corpus completo.

VIII. REFERENCIAS Y BIBLIOGRAFÍA

A. Referencias bibliográficas:

[1] O. Sposito, J. Bossero, E. Moreno, V. Ledesma, & L. Matteo. "Lexical Analysis Using Regular Expressions for Information Retrieval from a Legal Corpus", en *Computer Science – CACIC 2021*. Springer International Publishing, 2022.

[2] G. Kowalski. "*Information Retrieval Systems: Theory and Implementation*", 1st ed. Norwell, MA, USA: Kluwer Academic Publishers, 1997.

[3] C. Lorenzetti. "Caracterización Formal y Análisis Empírico de Mecanismos Incrementales de Búsqueda basados en Contexto". *Tesis Doctoral en Ciencias de la Computación* - Universidad Nacional del Sur. Buenos Aires, Argentina, 2011.

[4] G. Salton & M. Lesk. "Computer Evaluation of Indexing and Text Processing". J. *ACM*, 15(1): 8–36, 1968.

[5] P. Castells, M. Fernandez & D. Vallet. "An Adaptation of the Vector-Space Model for Ontology-Based Information Retrieval". *IEEE Transactions on Knowledge and Data Engineering*. 19(2): 261 – 272, 2007.

[6] J. Robredo. "Otimização dos processos de indexação dos documentos e de recuperação da informação mediante o uso de instrumentos de controle terminológico". *Ciência Da Informação*, 47(1). 2019. Disponible en: http://revista.ibict.br/ciinf/article/view/4431. Fecha de consulta: 07/02/22.

[7] G. Tolosa & F. Bordignon. "Introducción a la Recuperación de Información: Conceptos, modelos y algoritmos básicos". Universidad Nacional de Luján, Argentina, 2008. Disponible en: http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf. Fecha de consulta: 07/02/22.

[8] B. Harish & S. Guru & M. Shantharamu. "Representation and Classification of Text Documents: A Brief Review". *International Journal of Computer Applications, Special Issue on RTIPPR*. 1. 110 – 119, 2010.

[9] C. Sánchez Pérez. "Clasificación de Entidades Nombradas utilizando Información Global". *Tesis de Maestría*. Instituto Nacional de Astrofísica, Óptica y Electrónica. 2008. Disponible en: https://inaoe.repositorioinstitucional.mx/jspui/bitstream/1009/564/1 /SanchezPCR.pdf. Fecha de consulta: 06/03/2022.

[10] W. Gomaa & A. Fahmy. "A Survey of Text Similarity Approaches". *International Journal of Computer Applications*. 68(13), 2013.

[11] I. Amón, C. Jiménez. "Funciones de Similitud sobre Cadenas de Texto: Una Comparación Basada en la Naturaleza de los Datos", Disponible en: https://repositorio.unal.edu.co/bitstream/handle/unal/69915/71644758.20104.pdf?sequence=3&isAllowed=y. Fecha de consulta: 21/09/2022.

[12] A. Elmagarmid, P. Ipeirotis, & V. Verykios. "Duplicate Record Detection: A Survey". *IEEE Transactions on Knowledge and Data Engineering*, 19 (1): 1-16, 2007.

[13] R. Hamming. "Error detecting and error correcting codes". *The Bell System Technical Journal*; Vol. XXVI, No. 2, pp. 147-160, 1950.

[14] E. Gómez Ballester, "Aportaciones a la mejora de la eficiencia de la búsqueda del vecino más cercano", pp.5,19,137, [en linea], Fecha de consulta: 7/11/2022, https://rua.ua.es/dspace/bitstream/10045/28363/1/tesis_%20evagomezballester.pdf

[15] I. Amón, C. Jiménez, "Funciones de Similitud sobre Cadenas de Texto: Una Comparación Basada en la Naturaleza de los Datos", [en linea], Fecha de consulta:21/09/2022, https://repositorio.unal.edu.co/bitstream/handle/unal/69915/71644758.20104.pdf?sequence=3&isAllowed=y

[16] I.G. Albeniz, J.R. González de Mendivil, "Estudio sobre la detección de duplicados en orígenes de datos heterogéneos", [en línea], Fecha de consulta: 22/09/2022, https://academica-e.unavarra.es/xmlui/bitstream/handle/2454/16765/TF

G_Gorostizu_Albeniz_Ion.pdf;jsessionid=6C646114AECD758F433EF12200A60A92?sequence=1

B. Bibliografía:

C. Cardellino C., M. Teruel, L. Alonso Alemany, & S, Villata. "A Low-cost, High-coverage Legal Named Entity". 2017. Disponible en: https://hal.archives-ouvertes.fr/hal-01541446/document. Fecha de consulta: 28/10/2022.

M. Cucatto. "El lenguaje jurídico y su desconexión con el lector especialista: El caso de a mayor abundamiento." Letras de Hoje, 48 (1), pp. 127-138, 2013. Disponible en: http://www.memoria.fahce.unlp.edu.ar/art_revistas/pr.9102/pr.9102.pdf. Fecha de consulta: 06/8/2022.

C. Dozier, M. Light, A. Vachher, S. Veeramachaneni & R. Wudali. "Named Entity Recognition and Resolution in Legal Text". Semantic Processing of Legal Texts, pp.27-43, 2010.

Rodríguez Inés, P. El uso de corpus electrónicos para la investigación de terminologia jurídica (2008) Disponible en: https://www.tdx.cat/bitstream/handle/10803/286111/pri1de2.pdf?sequence=1. Fecha de consulta: 06/06/2022

V.I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals", Soviet Physics Doklady, pp.10:707–710, 1966. Disponible en: https://ui.adsabs.harvard.edu/abs/1966SPhD...10..707L/abstract Fecha de consulta: 08/11/2022

Recibido: 2022-11-18
Aprobado: 2022-12-23
Hipervínculo Permanente: https://doi.org/10.54789/reddi.7.2.4
Datos de edición: Vol. 7 - Nro. 2 - Art. 4
Fecha de edición: 2022-12-29

Lexical Analysis Using Regular Expressions for Information Retrieval from a Legal Corpus

Osvaldo Mario Spositto, Julio César Bossero, Edgardo Javier Moreno, Viviana Alejandra Ledesma(✉), and Lorena Romina Matteo

Department of Engineering and Technological Research, National University of La Matanza, Florencio Varela 1903, San Justo, La Matanza, Buenos Aires, Argentina
{spositto,jbossero,ej_moreno,vledesma,lmatteo}@unlam.edu.ar

Abstract. This article presents part of the work carried out in the framework of a research that aims to optimize an Information Retrieval System, by means of its specialization for the retrieval of legal documents. One of the fundamental sub-processes in this type of system is lexical analysis, in which indexing techniques are applied. These techniques involve extracting a series of concepts representative of the topics covered in a document, and then using them as access points for retrieval. This article describes a proposal for the extraction of information and identification of dates and references to named entities, such as File No., Resolution No., Article No. of Law XXX, which refer to the legal norm in force and are widely used in different judicial documents. For the recognition of such named entities, the process employed the definition of patterns using Regular Expressions, a way of representing a language in a synthetic form, applying a set of rules. From this, the terms obtained are stored in a matrix of terms/documents. This paper also describes the algorithms used during the validation of the proposed solution and presents the experimental results that show that by applying this method a significant reduction in the size of the inputs to the matrix can be achieved.

Keywords: Information retrieval systems · Regular expressions · Recognition of named entities · Lexical analysis

1 Introduction

This article is a continuation of the work presented at the 26th Argentine Congress of Computer Science, CACIC 2021, held in the city of Salta from 4 to 8 October 2021, organized by the Network of National Universities with Computer Science Degrees (RedUNCI) and the National University of Salta, under the title "*Propuesta para la construcción de un corpus jurídico utilizando Expresiones Regulares*" (Proposal for the construction of a legal corpus using Regulatory ExpressionsResults obtained) [1]. In this work, a theoretical proposal was presented to incorporate, in a legal corpus, references to dates and other common terms regularly used in the legal norm, by means of the Named Entity Recognition (NER) that make up the different judicial documents, using Regular Expressions (RE). RE are character strings that are used to describe or find patterns within other texts, using delimiters and syntax rules.

P. Pesado and G. Gil (Eds.): CACIC 2021, CCIS 1584, pp. 1–13, 2022.
https://doi.org/10.1007/978-3-031-05903-2_21

As expressed in the predecessor paper, jurisprudence has an important role as a source of law, because its conclusions support the application of the law in a specific case. The Argentinean judiciary produces every year a large number of rulings, files, among other things. These decisions are stored in documents, making this source of law ever larger, which drives professionals to spend more time to find relevant documents. Therefore, sophisticated computing techniques are needed to minimize search time and improve the relevance of retrieved documents.

The authors of this report belong to a group, which in the year 2021 presented a research project called "*Implementation of a Web System for Information Retrieval Oriented to Legal Documentation with the Parallelized Latent Semantic Indexing Process*", through the Incentive Program for Research Teachers of the Secretariat of University Policies (PROINCE).

Within the stages to carry out the aforementioned project, the theory and practice of documentary analysis, conceptual indexing, the development of a matrix of terms and the term/document matrix, which presents rows that correspond to terms and columns to documents, in this case a vector of documents is represented as a bag of words. In other words, to represent the textual content of the documents, this proposal uses a data structure consisting of a matrix with two dimensions, in which the indexing terms that have been extracted after processing the documents are stored in the system. These matrixes are widely used in the area of information retrieval, where the bag-of-words hypothesis captures to some extent the subject matter of the document [2].

As mentioned above, this article presents, on the one hand, the proposal to incorporate the references of both dates and the legal norm, through the NER or extraction of entities, such as Files, Decrees, Agreements, Articles, Laws, and others, that make up the different judicial documents, by means of patterns defined by RE. As an extension, an experimental test is also described in order to validate the proposed solution with the concrete results obtained.

2 Research Background

Regarding the work on corpus construction using RE to solve named entities, the work developed by Haag, in his thesis: "*Recognition of named entities in legal domain text*" [3], focuses on the detection, classification and annotation of named entities (e.g., Laws, Resolutions or Decrees) for the InfoLEG[1] corpus, a database that contains the documents of all the laws of the Argentine Republic. It should be noted that the search pattern presented in this article is based on the format presented by Haag, although it should be noted that his work does not include dates.

For his master's thesis, Duque Bedoya [4], presents a methodology to build a corpus for linguistic analysis, implementing a tagging system that includes the names of people, places, and dates by means of a computational tool. The information extraction processes include the automatic identification of such terms through the application of algorithms and heuristics used in digital libraries. The identification of events is carried out using the combination of the tags previously extracted from the corpus. The tool used for tagging

[1] http://www.infoleg.gob.ar/.

was Unstructured Information Management Architecture (UIMA) a free application to implement linguistic resources using JAVA and C++ programming languages, being compatible with the Eclipse programming environment.

Regarding the construction of various corpora, in the doctoral thesis of Rodriguez Inés [5], "*El uso de corpus electrónicos para la investigación de terminología jurídica*", an extensive list of the corpora available in Argentina and a detailed description on of more than 10 international multilingual corpora can be found. In addition, Cardellino's paper [6], "*A Low-cost, High-coverage Legal Named Entity*", can be mentioned. In this paper, an attempt is made to improve information extraction in legal texts by creating a legal named entity recognizer, classifier, and linker. The resulting tools and resources are open source and are aimed at developing a named entity recognizer, classifier and linker that exploits Wikipedia.

Another work worth mentioning is found in chapter two "*Regular Expressions, Text Normalization, Edit Distance*" of Jurafsky and Martin's book "*Speech and Language Processing*" [7], where a very clear explanation of the use of RE is given. In addition, a tool for performing language processing using RE is introduced in a theoretical way, and defines how to perform basic text normalization tasks, including word segmentation and normalization, sentence segmentation and lemmatization.

Finally, Robaldo et al., in their paper "*Compiling Regular Expressions to Extract Legal Modifications, present a prototype to automatically identify and classify types of modifications in Italian legal text*" [8]. This prototype uses XML language to define a new set of rules to identify the type of modifications in a text. The rule-based semantic interpreter implements a RE-based pattern matching strategy.

The proposal being presented in this paper is inspired by several aspects of the literature consulted and implements them in a process where a term matrix including the NER and date formats is created.

3 About Information Retrieval Systems

An Information Retrieval System (IRS) [2, 9, 10] is a tool that interacts between a corpus and its users. Its effectiveness depends on the adequate control of the language of representation of the information elements and the searches of its users. To meet its objectives, according to Tolosa et al. [2], an IRS must perform the following basic tasks:

- Logical representation of documents and, optionally, storage of the original.
- Representation of the user's need for information in the form of a query.
- Evaluation of the documents with respect to a query to establish the relevance of each one.
- Ranking of the documents considered relevant to form the "solution set" or response.
- Presentation of the response to the user.
- Feedback of the queries to increase the quality of the answer.

Robredo in [11], states that in any area of knowledge, meaningful terms can be used as descriptors to represent the content of written documents, in the processes of indexing and organizing information, as well as to formulate questions in the information retrieval process. Tolosa et al. in [2] state that the process can be divided into the following stages:

- Lexicographic analysis, words are extracted and normalized.
- Reduction (tokenization) of empty or high-frequency words.
- Lemmatization, words morphologically similar to a base or root form are reduced in order to increase the efficiency of a CRS.
- Selection of the terms to be indexed. Those simple or compound words that best represent the content of the documents are extracted.
- Assignment of weights or weighting of the terms that make up the indexes of each document.

With the advance of technology, the different communities generate an increasing volume of publications in different domains that are rapidly disseminated through different repositories. In this context, in order for users to find relevant publications for various purposes, the process of indexing these documents is the primary factor in achieving quality information retrieval and the consequent success of any search mechanism. Gil-Leiva in his paper explains why indexing is contextualized and provides a brief description of some of the most widely used automatic indexing systems [12]. With the above in mind, this work is framed in the lexicographic analysis within the process of an IRS. In this phase, the NER constitutes an independent tool for information extraction, which plays an essential role for a variety of applications related to natural language processing such as information retrieval.

4 Named Entity Recognition

According to [13] the term named entity "*...is a word or sequences of words that are identified as the name of a person, organization, place, date, time, percentage, or amount...*". Therefore, NER aims to recognize and classify such entities in various natural language processing applications. Several works have been reviewed detailing different uses of RE to detect patterns within the text of a document [3, 14, 15]. In the area of NER, a common problem is to obtain relevant information related to some of the mentioned entities, so it becomes important to be able to extract and distinguish this type of elements from the whole set of words that compose a document.

Although some elements are relatively easy to identify by using patterns, e.g., dates or numerical data, there are other elements, such as persons, places, or organizations , that present other difficulties to be identified as belonging to a specific type. In an

IRS, a technique such as NER is very important, as it allows searching for very specific information in collections of documents, extracting and organizing the relevant information. In the work of Sánchez Pérez [13], it is mentioned that in recent years there has been extensive work on the development of NER systems to improve the performance of classifiers using machine learning techniques.

As stated in [1], one of the factors influencing the success of standardization in computing has been the use of RE, a language for specifying text search strings [16].

5 Regular Expressions

Patterns constructed as RE allow the recognition of complexly structured character strings. Their name comes from the mathematical theory on which they are based. In chapter 3 of the book "*Introduction to the theory of automata, languages and computation*" [17], the authors argue that REs can be thought of as a "programming language", in which it is possible to write some important applications, such as text search or replacement applications. In fact, they are recognized by many programming languages, editors, and other tools. Therefore, ERs serve as the input language of many systems that process strings. Examples include the following:

1. Search commands such as the UNIX Grep[2] command or equivalent commands for locating strings in web browsers or text formatting systems. These systems employ an RE-like notation to describe the patterns the user wishes to locate in a file.
2. Lexical analyzer generators, such as Lex or Flex (a lexical analyzer is the component of a compiler that breaks down the source program into logical or syntactic units consisting of one or more characters that have a meaning). Logical or syntactic units include keywords (e.g., while), identifiers (e.g., any letter followed by zero or more letters and/or digits) and signs (e.g., '+' or '<=').

In other words, an RE is an algebraic notation for characterizing a set of strings. They are particularly useful for text search when we have a pattern and a corpus of texts to search. An RE search function will search the corpus and return all texts that match the defined pattern. From the guide in [18] the following example, explained in Table 1, has been developed: an RE can be used to check if an email is valid:

$$|"^[\\w-] + (\\.[\\w-]+)*@[A\text{-}Za\text{-}z0\text{-}9]+(\\.[A\text{-}Za\text{-}z0\text{-}9]+)*(\\.[A\text{-}Za\text{-}z]\{2,\})\$" \quad (1)$$

[2] https://www.gnu.org/software/grep/.

Table 1. Description of an RE to validate an e-mail.

Expression	Meaning
^	^ at the beginning of the RE, or at the beginning within the []
[\\w-] +	The + symbol indicates that one or more characters must appear within square brackets \\w indicates characters A to Z both upper and lower case, digits 0 to 9 and the symbol _
(\\.[\\w-] +)*	The * indicates that this group may appear 0 or more times. The email may optional-ly include a full stop followed by one or more of the characters in square brackets
@	The following must contain the @ character
[A-Za-z0–9]	In the RE after the @ must contain one or more characters that appear between the square brackets
(\\.[A-Za-z0–9] +)*	Followed (optionally, 0 or more times) by a full stop and 1 or more characters in square brackets
(\\.[A-Za-z] {2,})	Followed by a full stop and at least 2 characters appearing in square brackets
$	Marks the end of an RE

To perform this process, a program written in the C#[3] programming language is used and to solve the issue of RE, the REGEX library is used. C# is a programming language that is included in the.NET Platform and runs on the Common Language Runtime (CLR). The first language of importance for the CLR is C#, much of what is supported by the.NET platform is written in C#. This language is derived from C and C++, is modern, simple, and entirely object-oriented, and simplifies and modernizes C++ in the areas of classes, namespaces, method overloading and exception handling [19].

6 Test Development

This section presents the development of an experimental design test carried out in the context of the research. This aims to study the behavior of REs in the indexing process, in order to identify dates and named entities. The following paragraphs briefly describe the experiment carried out, followed by the methodology and also the parameters used. In order to carry out the tests, a corpus of 497 documents from a public body with similar characteristics, in terms of terminology used to legal documents was used (see Fig. 1).

Each of the documents has a structure similar to the one shown in Fig. 2, where dates in different formats can be distinguished, as well as other named entities. These make up the entries that are intended to be automatically identified and then normalised and incorporated into the repository, and finally used in the process of searching and retrieving the documents.

[3] https://docs.microsoft.com/en-us/dotnet/csharp/.

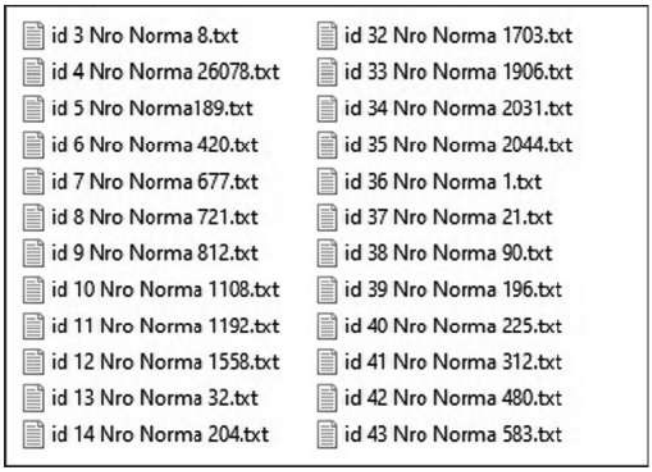


Fig. 1. Representation of the documents used in the experimentation.

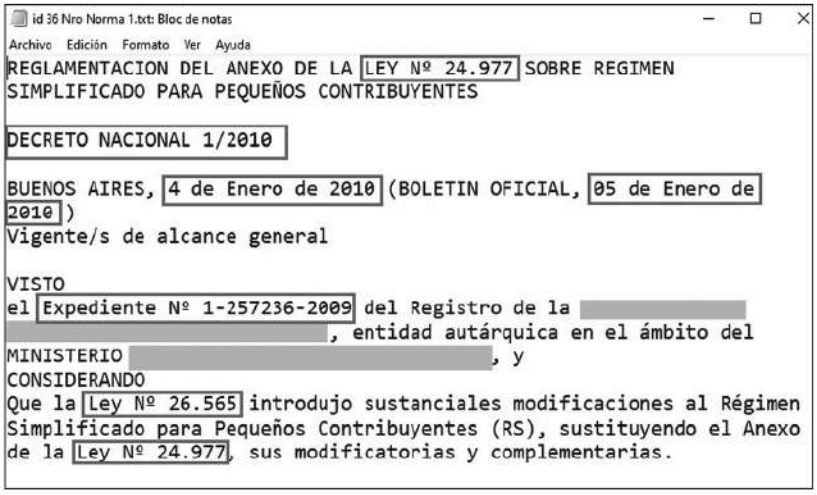


Fig. 2. Fragment of a document showing references to dates and named entities.

In agreement with [3] and also, based on the exploratory analysis carried out, the most common NER pattern is in the following form:

$$\text{<Entity Type>}\ [\text{Nro}]\ \text{or}\ [\text{N}^{\circ}]\ \text{<Number>}\ [/\text{<Year>}] \tag{2}$$

where "*Entity Type*" is a part of the named categories. In order to build a table of terms such as the one intended for this work, a common problem is to obtain relevant information related to all the names of the current judicial regulations to be standardised, so it becomes important to be able to extract and distinguish this type of elements from the whole set of words that make up a document.

In Fig. 3, a part of the pseudo code written in C# language and using the Regex library is shown. The following image shows how a breakpoint acts in the Visual Studio debugger. At this point the library recognizes in the selected paragraph, where a named entity is located. This identification is important to be able to normalise these names, and thus, to be able to incorporate them in a pre-established and uniform way.

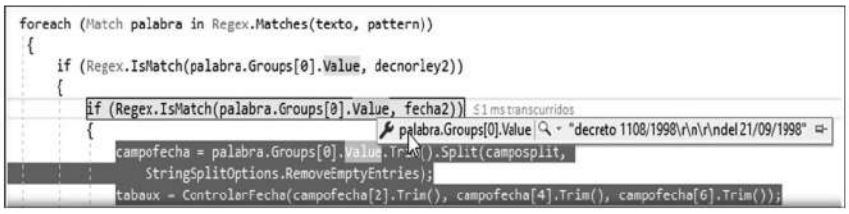


Fig. 3. Fragment of the pseudo code to identify named entities.

In relation to date references, which are found within text strings, there are a variety of applications available [21, 22] that help to convert different date formats using RE. Below is a collection of useful RE for finding dates in 'dd/mm/yyyy or yyyy' or 'dd-mm-yyy or yyyy' format:

AQ1

$$\textbf{RegEx1}: \ [0\text{-}9]\{1,2\}[0\text{-}9]\{1,2\}[0\text{-}9]\{2,4\} \text{ or} \tag{3}$$

$$\textbf{RegEx2}: \ \{1,2\}[\{1,2\}[0\text{-}9]\{1,2\}[0\text{-}9]\{2,4\} \tag{4}$$

Format 'Month, dd, ', e.g., '4 de julio de 2021'.

$$\begin{aligned}&(\text{Ene}(?:\text{ro})?|\text{Feb}(?:\text{rero})?|\text{Mar}(?:\text{zo})?|\text{Abr}(?:\text{il})?|\text{May}(?:\text{o})?||\text{Jun}(?:\text{io})?|\\ &\text{Jul}(?:\text{io})?|\text{Agost}(?:\text{o})?|\text{Sep}(?:\text{tiembre})?|\text{Oct}(?:\text{ubre})?|\\ &\text{Nov}(?:\text{iembre})?|\text{Dic}(?:\text{ciembre}) ?)\backslash s+(\backslash d\{1,2\})\backslash,\backslash s+(\backslash d\{4\})\end{aligned} \tag{5}$$

In Fig. 4, another fragment of the pseudo code is shown, in this case to identify a date format within a paragraph belonging to a document, using the Regex library.

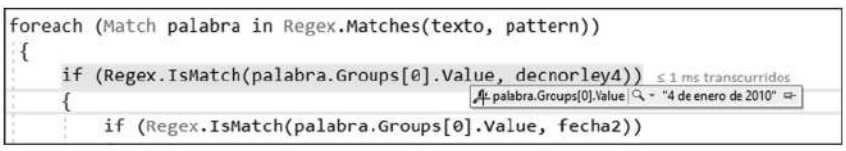


Fig. 4. Fragment of the pseudo code to identify date format labels with text.

The implemented indexing process can be summarized in the algorithm detailed in Table 2. Line 1 lists the inputs of the algorithm, where: D represents each of the documents to be processed that will be constantly executed, F the date REs, E the named entity REs, M is the term matrix and N is the maximum number of documents to

Table 2. Summary of the algorithm

Algorithm 1: Proposed method to obtain dates and named entities from a set of documents.

```
1:  Input: D, F, E, M y N.
2:  s ← 0;
3:  while s < M do
4:    for p • Ds do
5:      if p = F then
6:        TF ← get_date (p);
7:        NF ← normalize_date (TF);
8:      end if
9:      Matriz [s, k] ← valid_if_exists_F(NF);
10:     if p = E then
11:       TE ← get_entity (p);
12:       NE ← normalize_entity (TF );
14:     end if
15:     Matriz[s,k] ← valid_if_exist_E( NT);
16:   end for
17: end while
```

be processed. It is also necessary p, for each of the paragraphs composing a document D and the intermediate storages are represented by TF, NF, TE and NE.

The main loop occurs between lines 3 to 17, where the search and replace of the named dates and entities actually takes place. After being normalized and, after checking that they are only entered once in the matrix of terms, they are incorporated through a function.

7 Results Obtained

Throughout the project, some 6 REs were produced to identify some of the named entities for this experimentation and different date formats. Some of the REs made up covered more than one term to be searched for, as can be seen in Fig. 5.

```
String decnorley3 = "((E|e)xpediente|(E|e)xpte.|(L|l)ey|(D|d)ecreto|(N|n)orma)\\s*" + digito + "+";
```

Fig. 5. Fragment of the pseudo code to create an RE that finds the terms "Expediente", "Ley", "Decreto" or "Norma" (File, Law, Decree or Rule).

The document corpus contains a total of 1,279,205 terms. After running the whole process, it was possible to build the Table of Terms, whose structure is shown in Fig. 6.

The table contains 14,218 terms identified without repetition. In the figure, it can be seen that the program builds the table by assigning a unique ID to each term. Accompanying the term is the number of times it appears in the processed corpus. This value is used by other processes for the weighting of each term. This is not detailed in this paper, but it can be deepened in the presented bibliography [2, 9–11].

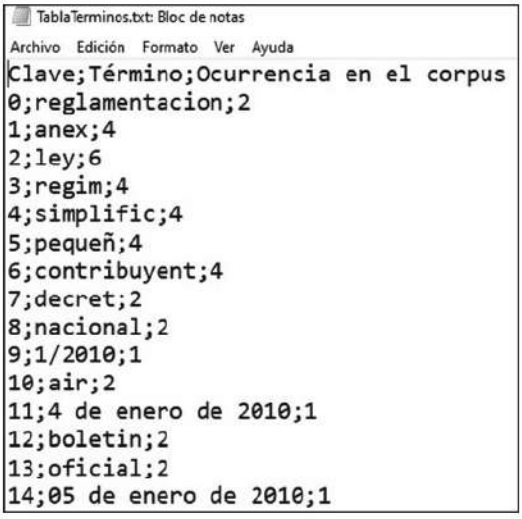


Fig. 6. Fragment of the Table of Terms with their respective frequencies.

Table 3 presents a summary of the entries found in the processing of the corpus used for the experimentation, including the main named entities detected.

Table 3. Summary of named entities and date formats in the selected corpus.

Reference	Example of the text that appears	Number of times
Art. XX	Art 14	817
arts. XX y XY	Arts. 14 y 17	76
artículo XX	artículo 125	10,404
artículo XX Inciso XXX	artículo 75 inciso 22	853
Ley XXXX o Ley N° XXX	Ley 26.660 o Ley N° 24.977	12,650
Expediente XX XXX o expte. XXX	Expediente N° 1-257236-2009 expte. 20–00160/2000	1,490
Decreto XXX/XXXX	Decreto 1/2010	427

On the other hand, Tables 3 and 4 show the results of the occurrences found in the indexing process, in this case, for date formats, both numeric and text.

Table 4. Summary of the ENs and date formats in the selected corpus

Reference	Number of times	Reference	Number of times
31/12/2015	3058	12/9/69	31
2/12/1964	91	6-11-2009	9
30/4/1970	700	3-1-2001	13
4/2/1971	409	03-10-64	92
31/12/70	24	2-21-94	4
1/02/90	8	20-5-97	1

In Tables 3, 4 and 5 it is possible to visualize the number of different formats that can be found for the same entities in a given corpus. With the algorithm proposed to be implemented, the aim is to bring all these different formats to a uniform one, for example, in the case of dates, to use a format composed of two digits for the day, two digits for the month and four digits for the year.

Table 5. Summary of some of the lettered date formats found.

Reference	Example of the text that appears	Number of times
XX de mes de XXXX	9 de febrero de 2009	1152
XX mes de XXXX	21 octubre de 2005	6
mes –XX	feb-01	48
Mes, XX de XXXX	Abril, 9 de 2007	158

8 Conclusions and Future Work

This paper presents an algorithm for searching and replacing text strings using RE. The performance for incorporating dates and named entities in a term indexing process is analyzed.

Through the experience carried out, it could be proved that a great advantage when applying RE to find named entities is that once the correct expression is defined, and after the corresponding exhaustive search, the entities that match exactly with that pattern will be all the existing ones in the corpus. This is all the more important because legal texts are very structured, and entities appear with a certain regularity.

In turn, the REs are a simple tool to use and do not require more than encoding the expression of the pattern itself, and not training a model for its recognition.

However, a possible disadvantage is their limitation in finding only the predefined patterns, so it is not possible to find another named entity that does not match any of the existing REs.

As is well known, it is of utmost importance to count on the participation of domain experts to validate the terms coming from the corpus, often by carrying out manual checks, which imply a considerable effort.

Thus, in order to reduce the manual intervention effort and to improve the performance in the exhaustive search for patterns, in a next stage, we intend to explore a complementary technique to the RE. This technique is known as Hamming Distance, a similarity metric, which allows us to reduce the dimensionality (number of terms) of the corpus.

As a next step, it is expected to test the algorithm proposed in this work by implementing it in the self-developed IRS. Furthermore, it is planned to build a legal corpus that will be used to evaluate the response times of the IRS and the relevance of the retrieved documents.

Acknowledgment. Thanks are due to the Department of Engineering and Technological Research of the National University of La Matanza, this work is financed within the framework of the PROINCE C241 project.

References

1. Spositto, O., et al.: Propuesta para la construcción de un Corpus Jurídico utilizando Expresiones Regulares. In: 26th Argentine Congress of Computer Science, CACIC 2021, pp. 746–755. National University of Salta, Buenos Aires (2021). http://sedici.unlp.edu.ar/handle/10915/129809. Accessed 25 June 2021
2. Tolosa, G., Bordignon, F.: Introducción a la Recuperación de Información: Conceptos, modelos y algoritmos básicos (2008). http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf. Accessed 25 June 2021
3. Haag, K.: Reconocimiento de entidades nombradas en texto de dominio legal (2009). https://rdu.unc.edu.ar/handle/11086/15323. Accessed 06 Jan 2022
4. Duque Bedoya, E.: Metodología para la Extracción de Metadatos Semánticos de Textos en español utilizando procesamiento de Lenguaje Natural: Subaplicación Para La Identificación De Contextos Espaciales Y Temporales En Textos Que Describan Interacciones Entre Actores. Universidad Eafit Departamento de Informática y Sistemas (2009). https://repository.eafit.edu.co/bitstream/handle/10784/1261/erika_duque_2009.pdf;jsessionid=19D87B68BAFF2D7E3D4296A8C4E727A4?sequence=1. Accessed 06 Jan 2021
5. Rodríguez Inés, P.: El uso de corpus electrónicos para la investigación de terminología jurídica (2008). https://www.tdx.cat/bitstream/handle/10803/286111/pri1de2.pdf?sequence=1. Accessed 06 Jan 2021
6. Cardellino, C., et al.: A low-cost, high-coverage legal named entity (2017). https://hal.archives-ouvertes.fr/hal-01541446/document. Accessed 06 Jan 2021
7. Jurafsky, D., Martin, J.: Speech and language processing (2020). https://web.stanford.edu/~jurafsky/slp3/2.pdf. Accessed 06 Jan 2021
8. Robaldo, L., et al.: Compiling regular expressions to extract legal modifications (2012). http://www.di.unito.it/~radicion/papers/robaldo12compiling.pdf. Accessed 06 Jan 2021

9. Kuna, H., Rey, M., Martini, E., Solonezen, L., Podkowa, L.: Desarrollo de un Sistema de Recuperación de Información para Publicaciones Científicas del Área de Ciencias de la Computación. Revista Latinoamericana de Ingeniería de Software, 107–114 (2014). http://revistas.unla.edu.ar/software/article/view/81. Accessed 06 Jan 2021
10. González, C.M.: La recuperación de información en el siglo XX. Revisión y aplicación de aspectos de la lingüística cuantitativa y la modeliza-ción matemática de la información (2008). http://www.fuentesmemoria.fahce.unlp.edu.ar/tesis/te.350/te.350.pdf. Accessed 25 June 2021
11. Robredo, J.: Otimização dos processos de indexação dos documentos e de recuperação da informação mediante o uso de instrumentos de controle terminológico. Ciência Da Informação **47**(1) (2019). http://revista.ibict.br/ciinf/article/view/4431. Accessed 25 June 21
12. Gil-Leiva, I.: SISA—automatic indexing system for scientific articles: experiments with location heuristics rules versus TF-IDF rules. Knowl. Organ. **44**, 139–162https://doi.org/10.5771/0943-7444-2017-3-139
13. Sánchez Pérez, C.: Clasificación de Entidades Nombradas utilizando Información Global (2008). https://inaoe.repositorioinstitucional.mx/jspui/bitstream/1009/564/1/SanchezPCR.pdf. Accessed 06 Jan 2022
14. Cucatto, M.: El lenguaje jurídico y su desconexión con el lector especialista: El caso de a mayor abundamiento. Letras de Hoje **48** (1), 127–138 (2013). http://www.memoria.fahce.unlp.edu.ar/art_revistas/pr.9102/pr.9102.pdf. Accessed 06 Jan 2021
15. Dozier, C., Kondadadi, R., Light, M., Vachher, A., Veeramachaneni, S., Wudali, R.: Named entity recognition and resolution in legal text. In: Francesconi, E., Montemagni, S., Peters, W., Tiscornia, D. (eds.) Semantic Processing of Legal Texts. LNCS (LNAI), vol. 6036, pp. 27–43. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12837-0_2
16. Seghiri, M.: Metodología protocolizada de compilación de un corpus de seguros de viajes: aspectos de diseño y representatividad. Rla. Revista de lingüística teórica y aplicada **49**(2), 13–30 (2011). https://doi.org/10.4067/s0718-48832011000200002. Accessed 06 Jan 2021
17. Hopcroft, J., Motwani, R., Ullman, J.: Introducción a la teoría de autómatas, lenguajes y computación. ISBN: 978-84-7829-088-8, p. 4. PEARSON Ed. S.A., Madrid (2007)
18. Stack Overflow Documentation: Aprendizaje de Expresiones Regulares. https://riptutorial.com/Download/regular-expressions-es.pdf. Accessed 06 Jan 2021
19. Cosio, L., Arrioja, N.: C#: Guía Total del Programador (2010). ISBN 978-987-26013-5-5
20. Regular Expression 101. https://regex101.com. Accessed 06 Jan 2021
21. RegEx Testing. https://www.regextester.com. Accessed 06 Jan 2021

ISBN 978-987-633-574-4

Propuesta para la construcción de un Corpus Jurídico utilizando Expresiones Regulares

Osvaldo Spositto[1], Ryckeboer Hugo[1], Viviana Ledesma[1], Gastón Procopio[1], Lorena Matteo[1], Cecilia Gargano[1], Julio Bossero[1], Edgardo Moreno[1], Victoria Saizar[1], Patricio Macias[1], Juan Ojeda[1], Fabio Quintana[1], Laura Conti[2], Sergio García[3] y Gustavo Pérez Villar[4]

[1] Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigación Tecnológicas. Florencio Varela 1903. San Justo. La Matanza. {spositto, hugor, vledesma, gprocopio, lmatteo, cgargano, jbossero, ej_moreno, vsaizar, pmacias, jmojeda}@unlam.edu.ar

[2] Universidad Nacional de La Matanza. Departamento Derecho y Ciencia Política. lconti@unlam.edu.ar

[3] Palacio de Tribunales. Departamento Judicial de Morón. Alte. Brown. Piso 4. Morón. sergiogabriel.garcia@pjba.gov.ar

[4] Subsecretaría de Tecnología Informática del Poder Judicial de la Provincia de Buenos Aires. Palacio de Justicia, avenida 13 entre 47 y 48, primer piso (La Plata). Argentina. gperez@scba.gov.ar

Abstract. En la última década, la construcción de corpus de distintas especialidades ha tenido un amplio desarrollo, debido en gran parte, por su incorporación en el proceso de recuperación de la información. Si bien, en el sistema jurídico argentino, existen varios buscadores de expedientes digitales, en este artículo se presenta una propuesta para incorporar, en un corpus jurídico, las fechas y las referencias de la norma jurídica, mediante el Reconocimiento de Entidades Nombradas (tales como Acordadas, Artículos, Leyes, entre otros), que componen los distintos documentos judiciales, utilizando Expresiones Regulares (ER). Estas cadenas de caracteres se utilizan para describir o encontrar patrones dentro de otros textos, empleando delimitadores y reglas de sintaxis. Se propone una metodología que permita identificar, clasificar y reemplazar estas entradas automáticamente, con el objetivo de ser normalizas. Por último, se presenta una propuesta para incorporar en un algoritmo de Lematización, la codificacion del proceso mencionado usando ER.

Keywords: Corpus, Expresiones Regulares, Sistema de Recuperación de Documento, Lematización, Reconocimiento de Entidades Nombradas

1 Introdución

Este trabajo, continúa con la línea de investigación y trabajo interdisciplinario entre especialistas del área jurídica provincial, técnicos de la Corte Suprema de la Provincia

de Buenos Aires e Investigadores de la Universidad Nacional de La Matanza (UNLaM). En el año 2020, el grupo abordó el análisis, diseño y construcción de una herramienta informática que ayuda a la sistematización y optimización de varios de los procesos judiciales que actualmente se realizan en forma manual o semiautomática en los juzgados de la provincia. La herramienta desarrollada, que se denomina *Experticia*[1] [1-2], pretende dar soporte a los operadores de la justicia en su decisión para la resolución de una causa. De esta manera se busca estandarizar el proceso de despacho de trámites, y a la vez agilizar y reducir los tiempos de carga, minimizando posibles errores como en el ingreso de datos. La información generada con Experticia, se almacena en el Sistema Informático de Gestión Asistida Multifuero (GAM), más conocido en el poder judicial como Augusta[2]. Este aplicativo fue creado con la finalidad de dotar al Poder Judicial de la Provincia de Buenos Aires, de una plataforma informática única e integral, que permita homogeneizar la gestión administrativa diaria de las causas. En el campo del derecho, la jurisprudencia tiene un papel importante como fuente de derecho; porque sus conclusiones apoyan la aplicación de la ley en un caso específico. El poder judicial argentino produce una gran cantidad de dictámenes, expedientes, etc. cada año, estas decisiones se guardan en documentos, haciendo que esta fuente de derecho sea cada vez mayor, lo que impulsa a los profesionales del derecho a dedicar más tiempo a la búsqueda de documentos relevantes. Por lo tanto, se necesitan técnicas sofisticadas de cómputo para minimizar el tiempo de búsqueda y mejorar la pertinencia de los documentos recuperados. Por este motivo, en el año 2021 se presentó el proyecto de investigación "*Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado*", por el Programa de Incentivos para Docentes Investigadores de la Secretaría de Políticas Universitarias (PROINCE). Dentro de las etapas para llevar adelante este trabajo, se encuentra la construcción de un corpus jurídico. Varias investigaciones se centraron en remarcar la importancia que tiene la lingüística de corpus como herramienta de ayuda para analizar terminología y fraseología especializada en su contexto original de producción. Hoy, gran parte de los corpus, se compilan a partir de textos electrónicos y la web se ha convertido en una gran fuente de contenidos textuales de todo tipo [3,4].

Un Sistema de Recuperación de Información (SRI) [5-7] es una herramienta que interactúa entre un corpus y sus usuarios. Su efectividad depende del adecuado control del lenguaje de representación de los elementos de información y las búsquedas de sus usuarios. Para cumplir con sus objetivos, según Gabriel H. Tolosa y otros [6], un SRI debe realizar las siguientes tareas básicas:

- Representación lógica de los documentos y – opcionalmente – almacenamiento del original.
- Representación de la necesidad de información del usuario en forma de consulta.
- Evaluación de los documentos respecto de una consulta para establecer la relevancia de cada uno.

[1] https://noficcionweb.com.ar/la-suprema-corte-bonaerense-y-la-unlam-avanzan-en-la-automatizacion-de-la-justicia/

[2] https://www.scba.gov.ar/paginas.asp?id=39889

- Ranking de los documentos considerados relevantes para formar el "conjunto solución" o respuesta.
- Presentación de la respuesta al usuario.
- Retroalimentación de las consultas (para aumentar la calidad de la respuesta).

La arquitectura de un SRI que permite realizar las tareas básicas enumeradas en el párrafo anterior se puede observar en la Figura 1:

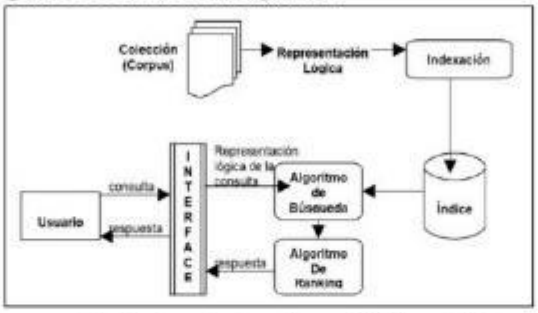


Fig. 1 – Arquitectura de un SRI. Fuente [6]

Como se puede apreciar en la Figura 1, el conjunto de todos los documentos sobre los que se deben realizar operaciones de RI se denomina corpus, colección de documentos o base de datos documental. El proceso de indexación genera la representación lógica de los documentos y las estructuras de datos denominadas índices, estas estructuras son las que permiten que se realicen búsquedas eficientes. El algoritmo de búsqueda se encarga de procesar la consulta de un usuario y de buscar en el índice cuáles documentos se asemejan a la consulta. A continuación, el algoritmo de ranking determina la relevancia de cada documento de acuerdo al nivel de semejanza y retorna un subconjunto con los documentos más relevantes. La interface de usuario permite que éste especifique la consulta, visualice la respuesta y realimente el sistema para mejorar la calidad de las respuestas.

Uno de los principales procesos de un SRI es la indexación y según Tolosa en [5] se puede dividir en las siguientes etapas:

- Análisis lexicográfico: Se extraen las palabras y se normalizan.
- Reducción (Tokenización) de palabras vacías o de alta frecuencia.
- Lematización: Se reducen palabras morfológicamente parecidas a una forma base o raíz, con la finalidad de aumentar la eficiencia de un SRI.
- Selección de los términos a indexar: Se extraen aquellas palabras simples o compuestas que mejor representan el contenido de los documentos.
- Asignación de pesos o ponderación de los términos que componen los índices de cada documento.

Si bien, estos corpus contienen información del mismo dominio, esta es habitualmente del tipo textual. En un expediente judicial, se pueden encontrar, además, referencias de fechas, en distintos formatos, como asi también, referencias a diferentes fuentes judiciales[3], como se puede ver en el párrafo siguiente: *"...de la Ley Nº 25.188, o el Decreto 41/99, o la Ley Nº 25.164 –que rige únicamente para el*

[3] https://www.conicet.gov.ar/wp-content/uploads/Ley-25164-De-Marco-de-Regulación-de-Empleo-Público-Nacional.pdf

personal.....su función (artículo 3° de la Ley N° 25.188; art. 47 del Decreto 41/99 y art. 30 de la Ley N° 25.164..."

Los usuarios de distintos ecosistemas, que utilizan corpus "*ad hoc*", demandan cada vez más servicios, que les permitan extraer información recuperada, usando reconocimiento y categorización de Entidades Nombradas (EN o NE del inglés Named Entity) de fácil integración en aplicaciones del Procesamiento del Lenguaje Natural (PLN) [8]. En este escrito, se presenta una propuesta, que se centra, en la detección, clasificación y normalización de fechas y entidades nombradas (como Acordadas, Artículos, Leyes, Resoluciones o Decretos, etc.) que componen la normativa jurídica, mediante el uso de Expresiones Regulares (ER). La idea es poder incorporar esta información al corpus, en el proceso de la Lematización de los documentos. Esta es una de las etapas de Preprocesamiento en un SRI [6]. Por su parte la técnica de Reconocimiento de EN (REN) se divide generalmente en dos pasos [8]: la delimitación de entidades nombradas y su posterior clasificación. En este trabajo solo nos enfocamos en la primera. Esta propuesta podría incrementar la eficacia en la equiparación entre los términos del documento y los términos de la pregunta del usuario.

2 Trabajos relacionados

Se han desarrollado muchos trabajos relacionados a la temática en cuestión. Diversas propuestas han sido consideradas para la construcción de Corpus jurídicos [9-10], en este último artículo, "*El uso de corpus electrónicos para la investigación de terminología jurídica*", se encuentra una extensa lista de los corpus disponibles en Argentina y una descripción detallada de mas de 10 corpus multilingües internacionales. Respecto a los trabajos sobre construcción de corpora utilizando Expresiones Regulares para resolver las Entidades Nombradas, tenemos el trabajo desarrollado por Karen Haag, en su tesis: "*Reconocimiento de entidades nombradas en texto de dominio legal"* [8], el escrito se centra en la detección, clasificación y anotación de entidades nombradas (como Leyes, Resoluciones o Decretos, entre otros) para el corpus de *InfoLEG*, una base de datos que contiene los documentos de todas las leyes de la República Argentina. Además, se pueden mencionar, entre otros, el trabajo de Cristian Cardellino [11] "*A Low-cost, High-coverage Legal Named Entity"*. En este documento, se intenta mejorar la extracción de información en textos legales mediante la creación de un reconocedor, clasificador y vinculador de entidad con nombre legal. Otro trabajo que merece ser nombrado se encuentra en el capítulo segundo: "*Regular Expressions, Text Normalization, Edit Distance*" del libro de D. Jurafsky y J. H. Martin: "*Speech and Language Processing*" [12], donde se presenta una herramienta para realizar tareas básicas de normalización de texto que incluyen segmentación y normalización de palabras, segmentación de oraciones y derivación. Por último, se puede nombrar, además, el trabajo de Robaldo, Livio y otros: "*Compiling Regular Expressions to Extract Legal Modifications*", que presenta un prototipo para identificar y clasificar automáticamente tipos de modificaciones en el texto legal italiano [13].

ISBN 978–987–633–574–4

3 Expresiones Regulares

Uno de los éxitos no reconocidos en la estandarización de la informática ha sido la utilización de ER, un lenguaje para especificar cadenas de búsqueda de texto [12]. Este lenguaje práctico se usa en todos los lenguajes de computadora, procesadores de texto y herramientas de procesamiento de texto como las herramientas Unix grep[4] o Emacs[5]. Formalmente, una expresión regular es una notación algebraica para caracterizar un conjunto de cadenas.

Son particularmente útiles para la búsqueda en textos, cuando tenemos un patrón y un corpus de textos donde buscar. Una función de búsqueda de expresiones regulares buscará en el corpus y devolverá todos los textos que coincidan con el patrón. El corpus puede ser un solo documento o una colección. Por ejemplo, la herramienta de línea de comandos de Unix grep toma una expresión y devuelve cada línea del documento de entrada que coincide con la expresión. En otras palabras, son notaciones simbólicas que se utilizan para identificar caracteres mediante una secuencia en el texto. En cierto modo, se parecen al método comodín del comando de Linux "Shell" para hacer coincidir los nombres de archivo y ruta, pero a una escala mucho mayor. Una expresión regular es un patrón capaz de reconocer o filtrar cadenas de caracteres según ciertos criterios. El uso de comodines "*" para indicar cadenas de caracteres cualesquiera o "?" para indicar un carácter único son ejemplos de uso de expresiones regulares. Así, el patrón "aba*" reconoce cadenas como "abaco", "abajo", "abatimiento", "abalorio", "aba-23"; el patrón "do?" reconoce cadenas como "doy", "dos", "dot", "don", "doS"; el patrón "aba*-txt" describe el conjunto de cadenas de caracteres que comienzan con "aba", contienen cualquier otro grupo de caracteres y luego la cadena "-txt". Los patrones construidos como ER que permiten reconocer cadenas de caracteres de estructura compleja. Las ER son utilizadas para realizar búsquedas o sustituciones en textos [14]. Estas son reconocidas por muchos lenguajes de programación, editores y otras herramientas. Su nombre proviene de la teoría matemática en la que se basan.

3.1 Expresiones Regulares básicas

Una ER determina un conjunto de cadenas de caracteres. Un miembro de este conjunto de cadenas se dice que aparea, equipara o satisface la expresión regular.

Con la idea de mostrar unos ejemplos, en la tabla 1, se pueden ver las ER que componen el conjunto de ER Elementales que aparean con un único carácter [14], en este mismo documento, se encuentra un tutorial del tema.

Tabla 1. Resumen de las ER Elementales que aparean con un único caracter [14].

Expresión	Aparea con
c	ER que aparea con el caracter ordinario c
.	(punto) aparea con un caracter cualquiera excepto nueva línea
[abc]	ER de un caracter que aparea con uno de a, b, c

[4] https://www.gnu.org/software/grep/
[5] http://www.gnu.org/software/emacs/

[^abc]	ER de un caracter que no sea uno de a, b, c
[0-9][a-z] [A-Z]	ERs de un caracter que aparean con cualquier caracter en el intervalo indicado El signo "-" indica un intervalo de caracteres consecutivos
\e	ER que aparea con alguno de estos caracteres (en lugar de la e): . * [\ cuando no están dentro de [ ] ^ al principio de la ER, o al principio dentro de [ ] $ al final de una ER / usado para delimitar una ER

Por lo general, se encontrará el nombre abreviado como "Regex" o "Regexp". En un editor de texto como EditPad Pro[6] o una herramienta de procesamiento de texto especializada como PowerGREP[7], puede usar la expresión regular como la siguiente:

«\b[A-Z0-9 ._%+-] + @ [A-Z0-9 .-] + \ . [AZ] {2,4} \ b » (1)

para buscar una dirección de correo electrónico. Cualquier dirección de correo electrónico, para ser exactos.

4 Reconocimiento de Entidades Nombradas

Encontramos en [15] una definición sobre el término Entidad Nombrada "*...es una palabra o secuencias de palabras que se identifican como nombre de persona, organización, lugar, fecha, tiempo, porcentaje o cantidad.*". El REN, tiene como objetivo el reconocer y clasificar nombres de personas, lugares, organizaciones o cantidades, en distintas aplicaciones del Procesamiento del Lenguaje Natural. A partir de la bibliografía consultada [8-11]. En estos trabajos se muestran distintos usos de ER para detectar patrones dentro del texto de un documento.

En el área del REN, un problema común es obtener información relevante relacionada con nombres de personas, lugares u organizaciones, por lo cual se vuelve importante el poder extraer y distinguir este tipo de elementos de todo el conjunto de palabras que componen a un documento. Aún cuando algunos elementos son relativamente fáciles de identificar, mediante el uso de patrones (por ejemplo: fechas o datos numéricos) existen otros elementos, como personas, lugares u organizaciones, que presentan otras dificultades para ser identificados como pertenecientes a un tipo específico. En un SRI, una técnica como el REN, es muy importante, ya que permite buscar información muy concreta en colecciones de documentos, extrayendo y organizando la información relevante [15]. En el trabajo de Sánchez Pérez, se menciona que en los últimos años se ha trabajado ampliamente en el desarrollo de sistemas de REN para mejorar el desempeño de clasificadores utilizando técnicas de aprendizaje automático.

[6] https://www.editpadpro.com/
[7] https://www.powergrep.com/

5 Bases metodológicas

Con el propósito de elaborar una propuesta orientada a la incorporación de datos en formato fecha y de entidades nombradas (Leyes, Acordadas, Decretos, Artículos, etc.), usando ER, en un corpus, utilizamos un texto público de Pedido de Libertad Condicional (PLC), disponible en [16]. En este documento se pueden observar cómo aparecen las referencias mencionadas en el texto (Fig.2).

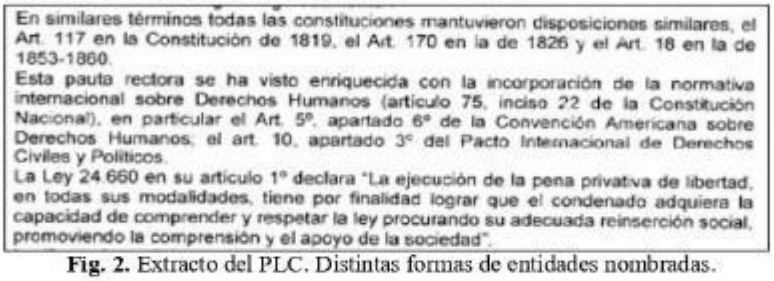

En similares términos todas las constituciones mantuvieron disposiciones similares, el Art. 117 en la Constitución de 1819, el Art. 170 en la de 1826 y el Art. 18 en la de 1853-1860.
Esta pauta rectora se ha visto enriquecida con la incorporación de la normativa internacional sobre Derechos Humanos (artículo 75, inciso 22 de la Constitución Nacional), en particular el Art. 5º, apartado 6º de la Convención Americana sobre Derechos Humanos, el art. 10, apartado 3º del Pacto Internacional de Derechos Civiles y Políticos.
La Ley 24.660 en su artículo 1º declara "La ejecución de la pena privativa de libertad, en todas sus modalidades, tiene por finalidad lograr que el condenado adquiera la capacidad de comprender y respetar la ley procurando su adecuada reinserción social, promoviendo la comprensión y el apoyo de la sociedad".

Fig. 2. Extracto del PLC. Distintas formas de entidades nombradas.

En el documento, usado de ejemplo, de casi nueve carillas, se puede contabilizar la cantidad de veces y distintos formatos, en que se encuentran estas referencias.

Tabla 2. Resumen de las EN y los formatos de fechas que figuran en el PLC.

Refencia	Ejemplo del texto que aparece	Cantidad de veces
Art. XX	Art 14	11
arts. XX y XY	arts 14 y 17	14
artículo XX	artículo 5	4
artículo XX Inciso	artículo 75 inciso 22	4
Ley XXXX	Ley 26.660	6
Fecha formato (dd/mm/aaaa)	24/11/1993	4
Fecha formato (dd de mes de aaaa)	27 de octubre de 2006	2
Fecha formato AAAA	*1993*	3

En coincidencia con [8] y también, en base a un análisis exploratorio del PLC, el patrón de REN más común, se encuentra en la siguiente forma:

< Tipo Entidad > [Nro] < Número > [/ < Año >] **(2)**

Dónde el "Tipo Entidad" es una parte de las categorías nombradas.

En la construcción de un corpus como el propuesto, para este trabajo, un problema común es obtener información relevante relacionada con todos los nombres de la normativa a normalizar, por lo cual se vuelve importante el poder extraer y distinguir este tipo de elementos de todo el conjunto de palabras que componen a un documento.

En el trabajo de Karen Haag [8], se desarrollan todas las entidades nombradas que se utilizan en el poder judicial. Algunos elementos son relativamente fáciles de identificar, mediante el uso de patrones (por ejemplo: fechas o datos numéricos). Existen muchas aplicaciones [17-18] que ayudan a convertir distintos formatos de fecha en ER. A continuación, se muestra una lista de algunas de las variantes que se pueden encontrar en este conjunto de datos:

- 20/04/2009; 20/04/09; 20/4/09; 3/04/09

- 20 de marzo de 2009; 20 de mar de 2009
- Febrero de 2009; septiembre del 09; octubre 2010
- 6/2008; 12/09 o 2009

A continuación, se muestra una colección de ER útiles para encontrar fechas:

- **Formato (dd/mm/aa o aaaa o dd-mm-aa o aaaa)**
  RegEx1: [0-9]{1,2}[\/-][0-9]{1,2}[\/-][0-9]{2,4} o **(2)**
  RegEx2: \d{1,2}[\/-]\d{1,2}[\/-]\d{2,4} **(3)**
- **Formato 'Mes, dd, aaaa', Por ejemplo, '4 de julio de 2021'.**
  (Ene(?:ro)?|Feb(?:rero)?|Mar(?:zo)?|Abr(?:il)?|May|Jun(?:io)?|Jul(?:io)?|Agost(?:o)?|Sep(?:tiembre)?|Oct(?:ubre)?|Nov(?:iembre)?|Dic(?:ciembre)?)\s+(\d{1,2})\,\s+(\d{4}) **(4)**

6 Lematización

En los SRI, la lematización (Stemming en Inglés) es una técnica empleada en la recuperación de datos en los SRI, que sirve para reducir variantes morfológicas de la forma de una palabra a raíces comunes o lexemas; con el fin de mejorar la habilidad de los motores de búsqueda para mejorar las consultas en documentos. Básicamente, este consiste en remover el plural, el tiempo, o los atributos finales de las palabras [19,20]. Cuando se realiza la extracción de palabras de un texto se obtiene una gran cantidad de entradas con formas verbales conjugadas y variantes de concordancia. Logrando la reducción morfológica de todas estas variantes se busca que el usuario recupere tanto los textos que contienen sus términos de búsqueda, como aquellos que contienen las formas derivadas de esos términos. Los algoritmos de lematización más conocidos son: Lovins[8](1968), Porter[9] (1980) y Paice[10] (1990). La descripción y comparación de estos y otros algoritmos menos conocidos, se encuentran desarrollados en el trabajo *"Comparative Study of Truncating and Statistical Stemming Algorithms"* en [21]. Todos eliminan "los finales" de las palabras en forma iterativa, y requieren de una serie de pasos para llegar a la raíz, pero no requieren "a priori" conocer todas las posibles terminaciones. Originalmente todos fueron hechos para el inglés, y se diferencian en la eficiencia del código y la elección de sufijos que identifican y eliminan. Esto es solo un ejemplo de la forma en que operan estos algoritmos. El trabajo de Porter[11], fue tomado como base por muchos investigadores [22]. El algoritmo[12] sirve para reducir variantes morfológicas de las formas de una palabra a raíces comunes o lexemas; mediante una sucesión de reglas que aplica sobre cada palabra. En esta memoria se presenta una codificación utilizando la librería Regex de .Net[13]. El ejemplo utiliza el método de *Regex.Replace* para reemplazar fechas con el formato mm/dd/aa por fechas con el formato dd-mm-aa.

[8] http://snowball.tartarus.org/algorithms/lovins/stemmer.html
[9] https://tartarus.org/martin/PorterStemmer/
[10] https://www.scientificpsychic.com/paice/paice.html
[11] https://tartarus.org/martin/index.html
[12] https://tartarus.org/martin/PorterStemmer/def.txt
[13] https://docs.microsoft.com/es-es/dotnet/standard/base-types/regular-expressions?redirectedfrom=MSDN

```
using System;
using System.Globalization;
using System.Text.RegularExpressions;
public class Class1
{   public static void Main()
   { string dateString =
DateTime.Today.ToString("d",
DateTimeFormatInfo.InvariantInfo);
      string resultString = MDYToDMY(dateString);
      Console.WriteLine("Converted {0} to {1}.",
dateString, resultString);    }
   static string MDYToDMY(string input)
   { try { return Regex.Replace(input,
     "\\b(?\\d{1,2})/(?\\d{1,2})/(?\\d{2,4})\\b",
           "${day}-${month}-${year}",
RegexOptions.None,
            TimeSpan.FromMilliseconds(150));     }
      catch (RegexMatchTimeoutException) {  return
input;}} }
```

7 Conclusiones y Trabajo Futuro

En este trabajo se propuso implementar, en un algoritmo de lematización, el uso de Expresiones Regulares para incorporar fechas y Entidades Nombradas a un corpus jurídico, para luego ser empleado en un Sistema de Recuperación de Información. Se estudiaron las expresiones regulares, que proporcionan un método eficaz y flexible para procesar texto.

Dentro de las tareas a desarrollar se puede mencionar:

- Incorporar la codificación propuesta al SRI implementado por el proyecto PROINCE mencionado en la introducción.
- Utilizar el algoritmo de Porter y analizar otros Lematizadores.
- Estudiar otras librerías existentes de ER.
- Realizar una clasificación de todas las EN dentro de la norma jurídica Argentina.

Referencias

1. Sposito O. y otros. Sistema Experto para Apoyo del Proceso de Despacho de Trámites de un Organismo Judicial. Jornadas Argentinas de Informática (JAIIO 2020).
2. Sposito O. y otros. Metodológica para evaluar un modelo de Justicia Predictiva". Trabajo presentado en CONAIISI 2020.
3. Capello, A. Sistema de recomendación para textos legales. (2018) En Linea: http://hdl.handle.net/11086/11342 Fecha de consulta: 25/6/21

4. Moreno A. Internet como fuente para la compilación de corpus jurídicos (2013) CES Felipe II (UCM) En línea: http://www.cesfelipesegundo.com/revista/Articulos2013/Art%C3%ADculoArsenioAndrade.pdf Fecha de consulta: 25/6/21
5. Kuna, H., Rey, M., Martini, E., Solonezen, L. & Podkowa, L. Desarrollo de un Sistema de Recuperación de Información para Publicaciones Cientificas del Área de Ciencias de la Computación, Revista Latinoamericana de Ingeniería de Software, (2014). 2(2): 107-114.
6. Tolosa G. & Bordignon, F. Introducción a la Recuperación de Información: Conceptos, modelos y algoritmos básicos. Universidad Nacional de Luján, Argentina, (2008). En línea: http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf. Fecha de consulta: 25/6/21
7. González, C. M. La recuperación de información en el siglo XX. Revisión y aplicación de aspectos de la lingüística cuantitativa y la modeliza-ción matemática de la información U. N. de La Plata. (2008) Disponible en: http:// www.fuentesmemoria.fahce.unlp.edu.ar/tesis/te.350/te.350.pdf. Fecha de consulta: 25/6/21
8. Karen Haag. Reconocimiento de entidades nombradas en texto de dominio legal. Cordoba, Argentina (2019). Recuperado el 01/08/2019 de: https://rdu.unc.edu.ar/handle/11086/15323
9. Cucatto M, El lenguaje jurídico y su desconexión con el lector especialista: El caso de a mayor abundamiento. Letras de Hoje, 48 (1), 127-138. (2013). En Memoria Académica. Disponible en: http://www.memoria.fahce.unlp.edu.ar/art_revistas/pr.9102/pr.9102.pdf Fecha de consulta: 25/6/21
10. El uso de corpus electrónicos para la investigación de terminología jurídica. Disponible en: http://www.bibliotecact.com.ar/PDF/08118.pdf. Fecha de consulta: 25/6/21
11. Cardellino C. y otros. A Low-cost, High-coverage Legal Named Entity. (2017) En: https://hal.archives-ouvertes.fr/hal-01541446/document. Fecha de consulta: 25/6/21
12. Jurafsky, D. & Martin, J. Speech and Language Processing. (2020) En línea:. https://web.stanford.edu/~jurafsky/slp3/2.pdf. Fecha de consulta: 25/6/21
13. Robaldo, L. y otros. Compiling regular expressions to extract legal modifications. 250. 133-141. 10.3233/978-1-61499-167-0-133. (2012).
14. William Shotts. The Linux Command Line. (Third Internet Edition). A LinuxCommand.org Book. (2016). En línea: https://filedn.com/liGIo7rEUfzfmU4MQdhIKrh/Cursos/CursoBasicoLinux/ExpresionesRegulares.pdf. Fecha de consulta: 25/6/21
15. Sánchez Pérez C. Clasificación de Entidades Nombradas utilizando Información Global. (2008). En línea: https://inaoe.repositorioinstitucional.mx/jspui/bitstream/1009/564/1/SanchezPCR.pdf. Fecha de consulta: 25/6/21
16. Revista Pensamiento Penal. http://www.pensamientopenal.com.ar/system/files/2016/06/miscelaneas43506.pdf#viewer.action=download. Fecha de consulta: 25/6/21
17. https://regex101.com/
18. https://www.regextester.com/
19 Martínez Méndez, F. Recuperación de información: modelos, sistemas y evaluación. Disponible en: https://digitum.um.es/digitum/bitstream/10201/4316/1/libro-ri.pdf. (2004) Último acceso: 20/07/2021.
20 Herrero Pascual, Cristina. (2010). Manual de indización: teoría y práctica. Investigación bibliotecológica, 24(52), 239-240. http://www.scielo.org.mx/scielo.php ?script=sci_arttext&pid=S0187-358X2010000300010&lng=es&tlng=es. Último acceso: 20/07/2021.
21 Figuerola C. y otros (2000) Diseño de un motor de recuperación de la información para uso experimental y educativo. Univ. de Salamanca. Disponible en: https://dialnet.unirioja.es/servlet/articulo?codigo=5555288. Último acceso: 20/07/2021.
22 Bordignon F., W. Panessi. Procesamiento de variantes morfológicas en búsquedas de textos en castellano. Revista Interamericana de Bibliotecología, ISSN 0120-0976, Vol. 24, Nº. 1 (ENE-JUN), 2001, págs. 69-88. https://dialnet.unirioja.es/servlet/articulo?codigo=4291340. Último acceso: 20/07/2021.

Código	FPI-009
Objeto	Guía de elaboración de Informe de avance de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

Implementación de un lematizador para la lengua española

Osvaldo Sposito[1], Hugo Ryckeboer[1], Julio Bossero[1], Edgardo Moreno[1], Viviana Ledesma[1], Gastón Procopio[1], Lorena Matteo[1], Cecilia Gargano[1], Victoria Saizar[1], Patricio Macias[1], Juan Ojeda[1], Fabio Quintana[1], Laura Conti[2], Sergio García[3] y Gustavo Pérez Villar[4]

[1] Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigación Tecnológicas. Florencio Varela 1903. San Justo. La Matanza.

{sposito, hugor, jbossero, ej_moreno, vledesma, gprocopio, lmatteo, cgargano, vsaizar, pmacias, fquintana, jmojeda}@unlam.edu.ar

[2] Universidad Nacional de La Matanza. Departamento Derecho y Ciencia Política. lconti@unlam.edu.ar

[3] Palacio de Tribunales. Departamento Judicial de Morón. Alte. Brown. Piso 4. Morón. sergiogabriel.garcia@pjba.gov.ar

[4] Subsecretaría de Tecnología Informática del Poder Judicial de la Provincia de Buenos Aires. Palacio de Justicia, avenida 13 entre 47 y 48, primer piso (La Plata). Argentina. gperez@scba.gov.ar

Resumen

El proceso de lematización se ha explorado ampliamente, sobre todo, por su aplicación en los Sistemas de Recuperación de Información (SRI). Se utiliza para la reducción de variantes de términos equivalentes semánticamente, llevándolos a una forma normalizada. Identifican un representante canónico para un conjunto de formas de palabras relacionadas. Para ello se eliminan las partes no esenciales de las palabras (sufijos, prefijos) para reducirlas a su parte original (lema). Esto mejora el proceso de indexar los documentos según su temática, ya que estos se agrupan por sus raíces. Las dificultades para desarrollar un algoritmo de este tipo es identificar y eliminar afijos, ya que cada idioma tiene características y reglas gramaticales únicas.

En este trabajo se modifica un lematizador, para el idioma español, basado en el algoritmo de Porter. Las modificaciones implementadas aumentan casi un 26% la obtención de lemas correctos.

Se pone a disposición la codificación de los algoritmos y los lotes de términos con sus respectivos lemas de prueba.

Contexto

El presente trabajo, es una continuación de los proyectos de investigación que se llevan adelante en la Universidad Nacional de La Matanza (UNLaM) del tipo PROINCE (Programa de Incentivos para Docentes Investigadores) de la Secretaría de Políticas Universitarias. Especialmente el trabajo titulado "Implementación de un Sistema de Recuperación de la Información", que se desarrolló en el período 2013-2014. Como resultado de este trabajo, se construyó íntegramente un prototipo de un Sistema de Recuperación de Información (en adelante SRI) con interfaz de aplicación de escritorio. Luego sobre la misma temática se realizaron, en orden cronológico, las siguientes investigaciones relacionadas con este tema:

"Optimización de la Recuperación de Documentos usando como Técnica base el ISL", período 2015-2016; "Uso de Minería de Datos para Acelerar la Recuperación de Documentos", período 2017-2018 y "Resolución Eficiente de la Descomposición en Valores Singulares en una Arquitectura Híbrida y su Posterior Inserción en un Sistema de Recuperación de Información", en el período 2019-2020.

1. Introducción

Si bien existen numerosas definiciones del término "*Recuperación de información*" (RI o IR, de *Information Retrieval* en inglés), algunos autores [1-2], coinciden en definirla como la aplicación de la tecnología informática para procesar información, incluye su adquisición, organización, recuperación y distribución. Por su parte el objetivo principal de un SRI, es básicamente organizar una colección de documentos, recibir consultas del usuario (que expresan textualmente su necesidad de información), procesarlas y devolver al usuario tandas de documentos empezando por los que su algoritmo considera más relevantes y un mínimo de documentos no relevantes (aquellos que, por la ineficacia del sistema, se consideran incorrectamente como relevantes, pero no de interés para el usuario) [3,4].

La RI, como se expresa en [5], tienen sus orígenes en las bibliotecas y centros de documentación en los que se requerían búsquedas bibliográficas de libros y artículos de revista. Debido a motivos históricos, los documentos en esos centros se representan utilizando un conjunto de términos índice o palabras clave. En la actualidad todavía existen fichas (manuales o electrónicas) en las que se rellenan los campos apropiados con esa información. En esos campos se incluyen datos como el título, autor, fecha de publicación, etc., del documento en cuestión. Pero también se incluyen otros términos que dan una indicación de su contenido, y que normalmente quedan reflejados en

el campo materia. Uno o varios especialistas asignan la materia de acuerdo con criterios más o menos subjetivos. Los usuarios que consultan el sistema de recuperación para buscar información deben traducir su necesidad informativa en una consulta adecuada al sistema de recuperación. Esto supone utilizar un conjunto de términos que expresen semánticamente su necesidad. En sistemas tradicionales también es habitual utilizar operadores booleanos para conectar varios criterios de búsqueda por campos diferentes. Como disciplina, la RI tiene pocos años, pero ha experimentado un rápido desarrollo, debido a la aparición de los motores de búsqueda. Un modelo de RI es la especificación sobre como representar documentos y consultas y cómo comparar unos y otros [6]. El objetivo de todo modelo es obtener un orden (ranking) de los documentos recuperados que refleje la relevancia de estos con la consulta del usuario

En el proyecto titulado "Implementación de un Sistema de Recuperación de la Información", se optó por el modelo algebraico, el propuesto por Salton y McGill [7], para ser utilizado en el filtrado, recuperación, indexado y cálculo de relevancia de documentos, al modelo del espacio vectorial. Una característica de este, es que cada documento, es representado mediante un vector de *n* elementos, siendo *n* igual al número de términos indizables que existen en la colección documental o corpus. Hay, pues, un vector para cada documento y, en cada vector, un elemento para cada término o palabra susceptible de aparecer en el documento. Cada uno de esos elementos es cubierto u ocupado con un valor numérico. Si la palabra no está presente en el documento, ese valor es igual a 0. En caso contrario, ese valor es calculado teniendo en cuenta diversos factores, dado que una palabra dada puede ser más o menos significativa (tanto en general como, sobre todo, en ese documento en concreto); este valor se conoce con el nombre de peso del término en el documento [5].

Básicamente, cualquier SRI se apoya principalmente en dos módulos: uno de indización, que construye los vectores de los documentos, y otro de consulta, que calcula la similaridad con una consulta dada [8]. Tanto los documentos como los vectores resultantes, así como productos intermedios y auxiliares, se almacenan en un repositorio auxiliar temporario. En el módulo de indización no todas las palabras o términos que componen un documento se incluyen en los índices. A los términos que se incluyen en el índice se les llama elementos de indización

En este sentido, es que se cree importante destacar brevemente, algunos aspectos generales que las técnicas de indización deben considerar. El primer punto es que no todas las palabras poseen el mismo nivel de significación para representar al documento. La teoría de la indización sugiere que algunas palabras conllevan más significado que otras [9]. Por ejemplo, los sustantivos más que los adjetivos o los verbos, y todas ellas más que las preposiciones. Otro aspecto es que incluir todas las palabras de un texto acarrea ruido en la recuperación. El concepto de indización implica un vocabulario seleccionado, representar al documento solo con lo más significativo. Por último, un tercer aspecto, es que el lenguaje natural presenta muchas variaciones, y que, al momento de buscar, es deseable expandir la búsqueda para incluirlas. Existen dos grandes grupos de variaciones lingüísticas. El caso en el que dos expresiones distintas cargan con significados muy similares: sinonimia, o justamente lo opuesto, cuando dos formas iguales tienen distinto significado: polisemia. Otro caso más complejo es cuando una frase textualmente igual puede ser interpretada de manera diferente según el contexto [10].

La mayoría de los SRI incluyen algún mecanismo que permite reducir el número de términos de indización utilizando algún control morfológico o de las formas flexionadas de las palabras. El concepto tomado del libro "*Introducción a la Morfología y Sintaxi*s" de Velma B. Pickett y Benjamin F. Elson [11define la morfología como la rama de la lingüística que estudia la estructura interna de las palabras para definir y clasificar sus unidades: las variantes de las palabras (morfología flexiva) y la formación de nuevas palabras (morfología derivativa y composición). Este tipo de mecanismo se utiliza considerando que aquellos términos con misma raíz tienen también un significado parcialmente equivalente. Por tal razón, estos procesos son utilizados para reducir considerablemente el universo de búsqueda sin que esto implique una pérdida importante de información. Eliminar en mayor o menor medida las variantes de un mismo lexema producidas por flexión: singular, plural, masculino, femenino, los tiempos verbales; y también las formas producidas por derivación: sufijos, prefijos, etc. El concepto de lexema, en lingüística, define la parte que se mantiene invariable en todas las palabras de una misma familia; expresa el significado común a toda la familia y puede coincidir o no con una palabra entera [11]. Estos algoritmos se aplican en el proceso de recuperación, de indización y en ambos.

Los SRI utilizan varias técnicas para realizar este proceso. Una de estas técnicas se conoce como *"conflación de términos"* [10], que es la reducción de la variedad lingüística de los documentos por medio de la agrupación de las ocurrencias textuales que se refieren a conceptos similares o idénticos. Esta técnica se ocupa de extraer los sufijos y prefijos comunes de palabras literalmente diferentes, pero con una raíz común, que pueden ser consideradas como un sólo término. Siguiendo a [12] encontramos que los algoritmos de conflación funcionan cortando el final o el comienzo de la palabra, teniendo en cuenta la lista de prefijos y sufijos comunes que se pueden encontrar en una palabra flexionada. Pero remarca que este corte indiscriminado puede tener éxito en algunas ocasiones, pero no siempre, por eso se considera que este enfoque presenta algunas limitaciones.

En [9,11,13] se señala que las técnicas más comunes de conflación son la truncación, la lematización o stemming y la aplicación de diccionarios. Para algunos autores estos dos términos (lematización y stemming) no son dos métodos exactamente iguales. Algunos autores [14,15 y 16] sugieren que la principal diferencia es la forma en que

funcionan y por tanto el resultado que devuelve cada uno de ellos. Para este trabajo, ambos conceptos, se desarrollan como sinónimos, ya que, en los SRI, estos tienen el mismo objetivo: reducir las formas flexivas de cada palabra en una base o raíz común. La lematización es un proceso lingüístico que, dada una palabra flexionada (ej. comiendo), encuentra su lema (ej. com). Una palabra está flexionada cuando está en plural (amigos), en femenino (amiga), conjugada (comiendo), en diminutivo (amiguita) o en superlativo (amigota grandota), etc. Harman en [6] afirma, que, en los SRI, la reducción de las palabras que tienen la misma raíz bajo el mismo término de indización puede incrementar la eficacia en la equiparación entre los términos del documento y los términos de la pregunta del usuario. La salida que obtendremos después de la lematización se llama "lema", que es una palabra raíz, después de la lematización, obtendremos una palabra válida que significa lo mismo [7].

El principal objetivo de este trabajo es proponer una mejora en el algoritmo de lematización utilizado en el proyecto 2013-2014, mencionado anteriormente. El código del programa original se encuentra disponible en la web[1] y está escrito en lenguaje PHP[2]. Para recodificar y aplicar las mejoras al algoritmo original y a la propuesta, se reescribieron los códigos en lenguaje C++. Para ello se empleó el entorno de desarrollo integrado o entorno de desarrollo interactivo, en inglés Integrated Development Environment (IDE) Code::Blocks[3], que es un software de código abierto, que soporta múltiples compiladores. La idea principal es comparar la calidad de los lematizadores, mediante distintas métricas los algoritmos en el proceso de extraer automáticamente los lemas de un lote de palabras.

2. Modelo conceptual de un SRI

Como ya se mencionó, en principio, la recuperación de información engloba las acciones encaminadas a identificar, seleccionar y acceder a los recursos de información útiles al usuario, el objeto documental se ha organizado y representado, utilizando una serie de normas y convenciones, en un soporte informático, mediante el diseño, creación y mantenimiento de bases de datos. Los SRI implementan una gama diversa de estructuras de datos, algoritmos y técnicas de recuperación de información, por ello, se precisa de un modelo conceptual donde se determinen: el tipo de almacenamiento, operaciones sobre los términos, modelos de búsqueda con base patrones exactos o los modelos inexactos los cuales contendrán las técnicas probabilísticas, los modelos lógicos y los espacios vectoriales [7]. En el trabajo de Martínez Méndez, se puede encontrar un estudio más profundo de los distintos modelos de RI existentes. Abordar en esta parte introductoria los diferentes modelos de SRI obedece a dos cuestiones principales. La primera es que se necesita exponer a nivel conceptual las ideas que han guiado a los experimentos de los cuales las técnicas de indización son parte. La segunda, de orden más práctico, es que ayuda a introducir al lector en los formalismos con que se expresan los procesos de RI para que su automatización sea posible. Una visión común en la RI es ver al documento y a la interrogación del usuario como contenedores de palabras que serán comparados, de manera que, cuantas más palabras en común tengan, más relevante será el documento para esa búsqueda.

Según Lorenzetti [17], en su tesis doctoral, escribió que los modelos de IR clásicos consideran que un documento está representado por un conjunto representativo de palabras claves, llamadas término índice, el autor asegura que un término índice es una palabra simple dentro de un documento, cuya semántica nos ayuda a recordar los temas principales sobre los que trata el documento, esta idea fue sugerida también por Luhn en los años 50s [18]. Entonces, dado un conjunto de términos de algún documento se puede notar que no todos son igualmente útiles a la hora de describir el documento. De hecho, hay algunos que son mucho más vagos que otros. No es un problema trivial decidir la importancia de un término como condensador del contenido de un documento. Más allá de esto, hay algunas propiedades de un término que son mensurables con facilidad y que son útiles para evaluar su potencialidad. Por ejemplo, consideremos una colección (o corpus) que contiene cien documentos. Una palabra que aparece en cada uno de los cien documentos es absolutamente inútil como término porque no nos dice nada acerca de cuáles documentos podrían interesarle a un usuario. Pero, otra palabra que aparezca en sólo cinco documentos sería más útil, porque reduce considerablemente el espacio de documentos que podría ser de interés para un usuario. Esto muestra que los distintos términos tienen una relevancia variable al usarlos para describir el contenido de los documentos. Siguiendo con Lorenzetti, también en [17], tomamos la definición que usa para la noción de peso de un término en un documento: *"...Sea un término k_i, un documento d_j y el peso asociado a (k_i, d_j), $w(k_i, d_j) \geq 0$. Este peso es una estimación de la importancia del término como descriptor del contenido semántico de un documento..."*.

En los SRI, varias tareas, como el almacenamiento, la búsqueda, agrupamiento o categorización de textos tienen como primer objetivo procesar documentos en lenguaje natural[4] (LN). Se entiende por LN a la técnica que es hacer que las máquinas comprendan los textos no estructurados y extraigan la información relevante de esos textos. El problema que surge es que los algoritmos que pretenden resolver estas tareas necesitan representaciones internas explícitas de los documentos. En el área de recuperación de información, como ya se comentó, una de las formas que se emplea es la expresión vectorial, donde las dimensiones del vector representan términos, frases,

[1] http://pragone.github.io/stemmer-es/

[2] https://www.php.net/

[3] https://www.codeblocks.org/

[4] https://www.sas.com/es_ar/insights/analytics/what-is-natural-language-processing-nlp.html#nlphowitworks

nombres propios o conceptos que aparecen en el documento.

2.1. Modelo de Espacio Vectorial

En este modelo, el texto es representado por un vector de términos, los términos comúnmente son palabras; cualquier texto puede ser representado por un vector en un espacio dimensional Salton en el año 1975 [19]. En el Modelo de Espacio Vectorial (MEV) los documentos se representan a partir de vectores, de la siguiente manera [20]:

$$\text{Vector } d_j = (w_{1j}, w_{2j}, w_{3j}, w_{4j}, \ldots, w_{nj}) \quad (1)$$

Donde ***n*** es igual al número total de elementos de representación considerados y por su parte ***w*** indica el peso que el término en concreto tiene para el documento ***j***, el peso de un término es una medida de su importancia en la representación del documento. Las comparaciones se realizan de acuerdo a lo siguiente: dos vectores pueden ser representados en el hiperespacio y pueden ser medidas las diferencias de dirección entre ambos de la misma forma que se haría en un espacio bidimensional mediante la comparación del coseno del ángulo que forman [2]. El producto escalar de dos vectores normalizados mide el coseno del ángulo que forman. Suficientemente para ordenarlos dada la monotonía existente entre ángulo y coseno. En la Figura 1 se muestra la representación del MEV para la comparación entre la consulta y los documentos.

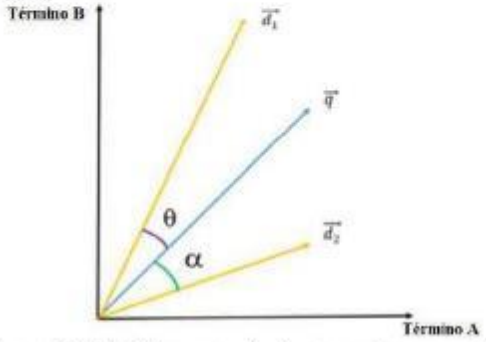


Figura 1. El MEV representa documentos y consultas como vectores para compararlos.

Para asignar una puntuación numérica a un documento para una consulta, este modelo mide la similitud entre el vector de consulta representado como vector q el vector del documento representado como vector d, típicamente el ángulo entre dos vectores es usado como una medida de divergencia entre los vectores, y el coseno del ángulo es usado como la similitud numérica [19]. El cálculo de la similitud cosenoidal, que es una medida que se calcula entre dos vectores distintos de cero, dentro del espacio interno del producto que mide el coseno del ángulo entre ellos. Esta se realiza mediante la siguiente ecuación [21].

$$SimCos(a,b) = \frac{\sum_{i=1}^{k} a_i * b_i}{\sqrt{\sum_{i=1}^{k} a_i^2} * \sqrt{\sum_{i=1}^{k} b_i^2}} \quad (2)$$

Se ve la conveniencia de tener los documentos representa_ dos por vectores normalizados y así evitar uno de los divisores en la fórmula. El otro divisor es constante y representa a la consulta, se lo puede omitir con lo cual quedan escalados los indicadores, pero no se pierde la monotonía entre evaluador y ángulo: A menor evaluador, mayor ángulo, o sea más disparidad de contenidos. Esta ecuación produce un valor entre 0 y 1, en donde un valor de 0 indica la perpendicularidad (ortogonalidad) de los vectores, lo cual indica que se tendrían dos vectores en dirección completamente diferente, y un valor de 1 indica que los vectores tienen una dirección idéntica [22].

2.1.1. Transformar un documento a valores numéricos

La transformación de un documento, que contiene palabras, a un vector de números con el cual los algoritmos pueden trabajar se realiza de la siguiente manera [23]. Supongamos estos tres documentos (cada frase es un documento):

1. Practica ajedrez.
2. Pesca en el rio.
3. Le gusta estudiar ajedrez.

La siguiente matriz tiene como filas a los documentos, y una columna por cada palabra diferente que hay en el total de documentos (vocabulario). La idea es poner la frecuencia de cada palabra en el documento. De esta manera el documento 1 tiene las palabras "practica", "ajedrez" por lo que la fila uno tiene valor 1 para esas palabras y 0 para el resto. En la Figura 2 se muestra un ejemplo de una matriz término-término, para el texto correspondiente al párrafo anterior, el contexto está formado por términos extraídos de un conjunto de oraciones.

	Practica	ajedrez	pesca	en	el	rio	le	gusta	estudiar
1	1	1	0	0	0	0	0	0	0
2	0	0	1	1	1	1	0	0	0
3	0	1	0	0	0	0	1	1	1

Figura 2. Matriz término-término del ejemplo anterior.

Solo con mirar las tres frases, uno diría que la 1 y la 3 son similares dado que ambas hablan de ajedrez. Sin embargo, si aplicamos la similitud coseno entre el 1 y el 3 nos da un valor de 0.258, recordemos que 0 es no se parecen nada y 1 los documentos son idénticos. Entre el 1 y el 2 nos da similitud = 0 y entre el 2 y el 3 también 0. Según el trabajo realizado en [24], las consecuencias de este tipo de normalización son:

1. El producto escalar entre dos vectores, el coseno del ángulo que los separa y la inversa de la distancia euclídea entre ellos, son funciones monótonamente crecientes entre sí, y, por lo tanto, equivalentes desde un punto de vista de la IR;

2. todos los documentos se consideran igualmente informativos; la diferencia entre documentos es así cualitativa (qué tipo de información contienen) y no cuantitativa (cuánta información contienen). Si bien es una posición filosófica debatible, la normalización ha probado dar buenos resultados.

Por lo que en este ejemplo tan simple parece que funcionan bien estas técnicas. Hay aun ciertas cosas que pulir en este método, para empezar, hay palabras que van a salir con frecuencia en muchos documentos. Los artículos (la, el, ella, ...) o preposiciones (a, de, por, ...) van a estar en gran cantidad en los documentos que analicemos. Realmente no nos dan ninguna información sobre el documento o similitud con otros, volviendo al ejemplo anterior, la palabra que era clave para la similitud era "ajedrez". Luego, se debería eliminar palabras superfluas que solo nos hacen ocupar espacio en la matriz. Este tipo de palabras se les denomina "*stopwords*" o "*palabras vacías*", que en RI, son las palabras que no tienen un significado por sí solas, sino que modifican o acompañan a otras, este grupo suele estar conformado por artículos, pronombres, preposiciones, adverbios e incluso algunos verbos. En el procesamiento de datos en lenguaje natural son filtradas antes o después del proceso en sí, no considerándolos por su nulo significado, en el caso de los buscadores como Google no lo consideran al momento de posicionar, pero si al momento de mostrar los resultados de búsqueda. Antes de empezar a trabajar con los documentos conviene eliminarlas de los documentos, para reducir la cardinalidad de la matriz. Por último, luego de este proceso se debe aplicar otro para quedarse sólo con la raíz de la palabra, como se comentó en los apartados anteriores.

3. Lematización

Como ya se comentó, la lematización es una técnica empleada en la recuperación de datos en los SRI, que sirve para reducir variantes morfológicas de la forma de una palabra a raíces comunes o lexemas; con el fin de mejorar la habilidad de los motores de búsqueda y, a consecuencia, los resultados de las consultas. Básicamente, este consiste en remover el plural, el tiempo, o los atributos finales de la palabra [6,20]. Cuando se realiza la extracción de palabras de un texto se obtiene una gran cantidad de entradas con formas verbales conjugadas y variantes de concordancia. Logrando la reducción morfológica de todas estas variantes se busca que el usuario recupere tanto los textos que contienen sus términos de búsqueda, como aquellos que contienen las formas derivadas de esos términos. Encontramos en [10] este ejemplo, en idioma inglés, "analysis", "analyzing", "analyzer", "analysing" puede reducirse a la forma "analy" que se considera su raíz. Los plurales, los gerundios y las formas de los verbos en pasado son los casos más comunes de palabras susceptibles de aplicar esta técnica. Durante el proceso de lematización siempre existirá un porcentaje de error, pero este es lo suficientemente bajo como para no afectar a la efectividad en la recuperación.

Sin embargo, debido a que este proceso generalmente se basa en heurísticas, está lejos de ser perfecta. De hecho, los errores más frecuentes tienen que ver con la "sobre-reducción" y la "sub-reducción". El primero se produce cuando se corta demasiada palabra. Esto puede resultar en raíces sin sentido, donde todo el significado de la palabra se pierde o se confunde. O puede resultar en que las palabras se resuelvan con las mismas raíces, aunque probablemente no deberían serlo. Tome las cuatro palabras "*universidad*", "*universo*", "*universidades*" y "*universos*". Un algoritmo que resuelve estas cuatro palabras en la raíz "*univers*" se ha sobre-reducido. Si bien podría ser bueno que el "universo" y los "*universos*" se fusionen y la "*universidad*" y las universidades se fusionen, los cuatro no encajan. Una mejor resolución podría hacer que los dos primeros se decidan a "*univers*" y los dos últimos se decidan a "*universi*". Pero hacer cumplir las reglas que lo establezcan podría dar lugar a que surjan más problemas. La sub-reducción, por otra parte, es el problema opuesto. Viene de cuando tenemos varias palabras, que tienen el mismo lema y significan cosas distintas. Esto se puede ver si tenemos un algoritmo de derivación que deriva las palabras *"salida", "sala"* y *"salada"* en el mismo lema: "*sal*". [10].

Los algoritmos de lematización más conocidos son: Lovins[5] (1968), Porter[6] (1980) y Paice[7] (1990). La descripción y comparación de estos y otros algoritmos menos conocidos, se encuentran desarrollados en el trabajo "*Comparative Study of Truncating and Statistical Stemming Algorithms"* en [8]. Estos algoritmos, tiene en común que eliminan "*los finales*" de las palabras en forma iterativa, y requieren de una serie de pasos para llegar a la raíz, pero no requieren "*a priori*" conocer todas las posibles terminaciones. Originalmente todos fueron hechos para el inglés, y se diferencian en la eficiencia del código y la elección de sufijos que identifican y eliminan.

Según la bibliografía consultada, una de las formas más utilizadas de lematización, es la eliminación de afijos de palabras, la construcción de este proceso depende en gran medida del idioma para el que se desarrolló [20]. Por ejemplo, como se detalla en [25], el autor remarca que: en el inglés, las flexiones de los verbos son apenas 4 por verbo, los sustantivos y adjetivos solamente existen en dos formas (plural y singular), los artículos son pocos y no poseen género ni número. En cambio, en el español, solamente la flexión de los verbos involucra: 3 personas, 2 números 4 tiempos, 3 modos y ciertas combinaciones especiales, resultando cerca de 70 formas verbales simples, si a esto anexamos las formas compuestas y los pronombres enclíticos anidados, tenemos varias centenas de posibles palabras y grupos a considerar por cada verbo. Con los sustantivos y adjetivos, suceda algo similar si consideramos los sufijos y prefijos diminutivos, aumentativos, peyorativos y otros, además del plural, singular y algunas veces, el neutro, el colectivo (para grupos). Un diccionario completo para el español que contenga las formas más usadas, tendría más de 93000 lemas según la 23.ª edición del diccionario de la RAE de 2014[8].

---

[5] http://snowball.tartarus.org/algorithms/lovins/stemmer.html

[6] https://tartarus.org/martin/PorterStemmer/

[7] https://www.scientificpsychic.com/paice/paice.html

[8] https://www.rae.es/obras-academicas/diccionarios/presentación-del-diccionario-de-la-lengua-espanola-y-sus-ediciones

3.1. El algoritmo de Porter

El trabajo de Porter[9] fue tomado como base por muchos investigadores. El algoritmo[10] lee un archivo, toma una serie de caracteres y de esa serie, una palabra; luego la valida verificando que todos los caracteres involucrados sean letras, de ser así, aplica la lematización sobre ella. Esta consiste en hacer pasar esta palabra a través de varios conjuntos de reglas, en [26] se explica detalladamente el proceso, en resumen, se puede decir, que cada conjunto de reglas está formado, por *n* reglas y cada regla por:

1. Un identificador de regla
2. El sufijo a identificar
3. El texto por el cual debe ser reemplazado al encontrar el sufijo
4. El tamaño del sufijo
5. El tamaño del texto de reemplazo
6. El tamaño mínimo que debe tener la raíz resultante luego de aplicar la regla (esto es a los efectos de no procesar palabras demasiado pequeñas)
7. Una función de validación (una función que verifica si se debe aplicar la regla una vez encontrado el sufijo)

El algoritmo se desarrolló para la lematización de textos en inglés, pero la creciente importancia de la recuperación de información en la década de 1990 llevó a una proliferación del interés en el desarrollo de técnicas de combinación que mejorarían la búsqueda de textos escritos en otros idiomas [25]. Por lo tanto, el algoritmo proporcionó un modelo para el procesamiento natural de textos que no estén en inglés. Porter ha desarrollado toda una serie de lematizadores que se basan en su algoritmo original y que cubren las lenguas romances (francés, italiano, portugués y español), las germánicas (holandés y alemán) y escandinavas (danés, noruego y sueco), así como finlandés y ruso. El autor demostró cómo su algoritmo de supresión de sufijos mejora frente a otros sistemas más complejos la ejecución de la recuperación en términos de exhaustividad.

Este algoritmo tiene dos consideraciones importantes, una es que el sufijo que se elimine sea siempre el más largo y que la palabra cortada mantenga una determinada longitud. Teniendo esto en cuenta, cada palabra puede ser lematizada tantas veces como se considere necesario. El algoritmo, resumido en el trabajo de [27], se fundamenta en:

La medida (m) de la raíz, se basa en la alternancia de vocales (a, e, i, o, u), y (en el caso de que vaya precedida de una consonante) y consonantes, (todas las letras que no son vocales). En su artículo original, Martin F. Porter, que presentó en la Universidad de Cambridge, analiza explícitamente el uso del algoritmo para los sistemas de recuperación de información. El algoritmo, para la lengua inglesa, procede como sigue:

$$[C]\ (VC)^m\ [V]$$

Donde:

[C]: Consonante susceptible de aparecer
(VC): Conjunto de vocal/consonante
m: medida de cada palabra o parte de palabra
[V]: vocal susceptible de aparecer
Cuando m es igual a 0 la palabra es nula.
Condiciones de la raíz
<x> la raíz termina con la letra x

- v la raíz contiene una vocal
- d la raíz termina en doble consonante
- la raíz termina con una secuencia del tipo consonante-vocal-consonante, donde el final de la consonante no es w, x, o y

Estas condiciones se pueden combinar entre sí y con la longitud de *m* mediante operadores booleanos.

Reglas [27]:

- (condición)S1→ S2
  Lo que significa que el sufijo S1 se reemplaza por S2 si las letras restantes de S1 satisfacen la condición.
- El primer paso del algoritmo está diseñado para tratar con participios y plurales pasados.
- Este paso es el más complejo y está dividido en tres partes en la definición original, 1a, 1b y 1c.
- La primera parte trata de los plurales, por ejemplo, sses → ss y la eliminación de s.
- La segunda parte elimina ed y ing o realiza eed → ee cuando sea apropiado. La segunda parte continúa solo si se elimina ed o ing y transforma el tallo restante para garantizar que se reconozcan ciertamente suficientes más adelante.
- La tercera parte simplemente transforma una terminal y en una i, esta parte se inserta como el paso 2.

Se puede encontrar una implementación del algoritmo de Porter en el portal de NLTK[11] (Natural Language Toolkit), que es una plataforma líder para crear programas Python que funcionen con datos de lenguaje humano.

Proporciona interfaces fáciles de usar para más de 50 corpus y recursos léxicos como WordNet, junto con un conjunto de bibliotecas de procesamiento de texto para clasificación, tokenización, derivación, etiquetado, análisis y razonamiento semántico, códigos para bibliotecas de procesamiento del lenguaje natural (NL) de nivel industrial, y un foro de discusión activo.

Se observa que los algoritmos de Lematización dependen en gran medida del idioma en el que están escritos los documentos. Este algoritmo ha sido ampliamente utilizado, referenciado y adaptado durante las últimas tres décadas. Varias implementaciones del algoritmo están disponibles en la WEB, incluido el sitio

[9] https://tartarus.org/martin/index.html
[10] https://tartarus.org/martin/PorterStemmer/def.txt
[11] https://www.nltk.org/

web oficial[12] escrito y mantenido por el autor para la distribución de su algoritmo.

3.2. El algoritmo de Snowball

Una modificación del algoritmo trabajo de Porter, es el algoritmo de Snowball[13]. Este puede mapear palabras que no están en inglés.

Dado que es compatible con otros idiomas, el algoritmo puede denominarse lematizador multilingües. El proyecto Snowball, tiene un portal de donde descargarse los códigos fuentes para distintos idiomas: lenguas romances[14] (francés, italiano, portugués y español), las germánicas[15] (holandés y alemán) y escandinavas[16] (danés, noruego y sueco), así como finlandés y ruso[17].

Este algoritmo, fue el implementado en el SRI del año 2013-2014. El mismo, como se comentó, siguió los pasos que propuso Porter.

Este algoritmo funciona dividiendo la palabra en dos regiones, R1 y R2, sin embargo, en algunos casos también se usa otra región RV. R1, R2 y RV se pueden definir de la siguiente manera:

- R1 - región que se encuentra después de la primera letra no vocal (consonante o letras acentuadas) después de una vocal;
- R2 - es la región después de la primera no vocal después de una vocal en R1, o es la región nula al final de la palabra si no hay vocal.
- RV - si la segunda letra es una consonante, RV es la región después de la siguiente vocal siguiente, o si las dos primeras letras son vocales, RV es la región después de la siguiente consonante, de lo contrario (consonante-vocal) RV es la región después de la tercera letra.

Pero RV es el fin de la palabra si no se cumplen esas condiciones. En la Figura 3 se presenta un ejemplo, de nuestra aplicación, de las regiones R1, R2 y RV de la palabra "*universidad*".

En esta imagen se puede ver al final de la misma, el lema resultante: "*univers*"

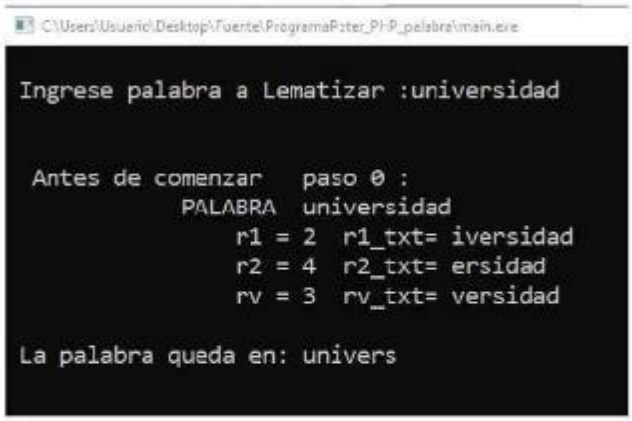


Figura 3. Salida del lematizador modificado.

4. Metodología utilizada

Para llevar a cabo la experiencia se seleccionaron 27.265 palabras. Estas fueron elegidas al azar de un lemario que en total posee 1.113.014 términos. Esta colección fue construida por el Dr. Ignacio Mario Morales Flores[18], para su Corrector Ortográfico para Medicina, el mismo es compatible con Word 2016 de Microsoft.

Como se comentó anteriormente, para el SRI desarrollado en el año 2014, se empleó el lenguaje de programación C# (Sharp)[19], para recodificar el proceso de lematización original del algoritmo de Snowball. En esta oportunidad, el código fue reescrito en lenguaje C++ y se encuentra a disposición para el lector, al igual que el resto de los archivos empleados en esta experimentación en [28] se deja un correo para solicitar los siguientes archivos:

- Lematizador_x_palabra.rar: Código fuente que devuelve el lema de la palabra ingresada. Muestra el detalle de cada paso del algoritmo.
- Lematizador_modificado.rar: Código con las propuestas incorporadas. Este programa tiene fijos o hardcodeados los datos a procesar.
- LotePrueba.csv: Archivo con palabras y sus lemas.
- verbos.txt: Listado de los verbos 9281 verbos. Obtenidos del mismo sitio que el lemario.

Para obtener los lemas de las 27.265 palabras seleccionadas, se empleó un lematizador para la lengua española que se encuentra disponible en el sitio: *https://snowballstem.org/demo.html#Spanish*. En la fig. 4 se muestra la pantalla principal del mismo. Este conjunto de datos, se tomó para la experimentación, como los lemas correctos a comparar con los lemas devueltos por los algoritmos utilizados en el SRI.

[12] https://tartarus.org/martin/PorterStemmer/
[13] https://snowballstem.org/demo.html
[14] https://snowballstem.org/algorithms/romance.html
[15] https://snowballstem.org/algorithms/germanic.html
[16] https://snowballstem.org/algorithms/scandinavian.html
[17] https://snowballstem.org/algorithms/

[18] http://www.cpimario.com
[19] https://docs.microsoft.com/en-us/dotnet/csharp/

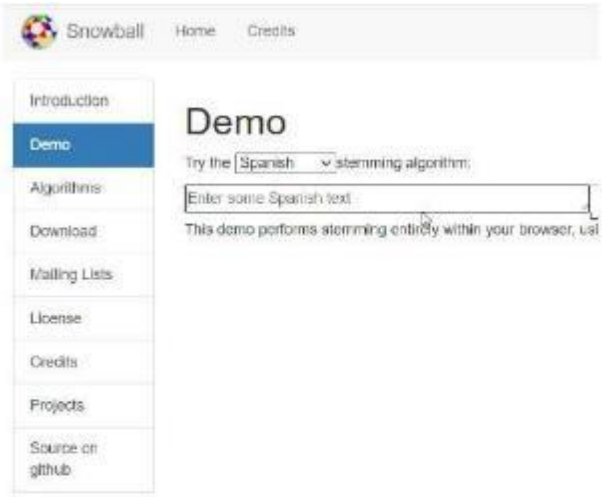


Figura 4. Pantalla del lematizador.

4.1. Evaluación de los algoritmos

Para evaluar los algoritmos de lematización, tanto el original como el modificado, se tomó como guía, el trabajo presentado por Gurusamy, Kannan y Nandhini [27], que evaluó el análisis de rendimiento de los algoritmos básicos de eliminación de tres sufijos derivados en el idioma inglés llamados Lovins, Porter y Paice / Husk mediante la precisión y distintas métricas de desempeño de los algoritmos. En el trabajo mencionado se evalúan los siguiente dos parámetros:

A. Precisión del algoritmo.
B. Perfomance o rendimiento del algoritmo.

A. Precisión del algoritmo de derivación

Se basa en el número de palabras derivadas correctamente dado por los algoritmos de lematización y el número de únicas palabras en los conjuntos de datos dados. La exactitud es calculada por el debajo de la fórmula:

Precisión = (Nº palabras correctamente derivadas / Nº de palabras únicas) * 100 **(3)**

B. Rendimiento del algoritmo de derivación.

Se han utilizado cinco métricas de la siguiente manera:

1. Media de palabras por clase de combinación. Este es el número promedio de palabras que corresponden a la misma raíz o lema para un corpus. Por ejemplo, si las palabras *"ingeniero"*, *"ingenieril"* e *"ingeniería"* se derivan de *"ingeniero"*, Entonces el tamaño de esta clase de combinación es de tres.

Los lematizadores más fuertes tenderán a tener más palabras por clase de combinación. Si la combinación de 1000 palabras diferentes da como resultado 250 raíces distintas, luego el número medio de palabras por la clase de fusión sería 4. Obviamente, esta métrica depende del número de palabras procesadas, pero para una colección de palabras de un tamaño determinado, un valor más alto indica una lematización más pesada. El valor se calcula de la siguiente manera:

$$MWC = N / S \quad (4)$$

Donde **MWC** es el número medio de palabras por combinación clase.

N: Número de palabras únicas antes de la derivación.
S: Número de lemas únicos después de la derivación.

2. Factor de compresión del índice. Es la reducción fraccional en tamaño del índice logrado mediante la lematización. Por ejemplo, un corpus con 50.000 palabras (N) y 40.000 raíces (S), Tendría un factor de compresión del índice del 20%. El mejor lematizador tenderán a tener factores de compresión de índice más grandes. Esto se puede calcular mediante:

$$ICF = (N-S) / N \quad (5)$$

Donde **ICF** es el factor de compresión del índice

N: Número de palabras únicas antes de la derivación
S: Número de lemas únicos después de la derivación

3. El número de palabras y raíces que difieren (NPRD). La lematización a menudo deja las palabras sin cambios. Por ejemplo, una derivación podría no alterar "*reacción*" porque ya es una palabra raíz. Los lematizadores más fuertes cambiarán las palabras con más frecuencia que los lemas más débiles.

4. Media de caracteres eliminados al formar el lema (MCE): Hay varias métricas que utilizan el principio de que un lematizador fuerte eliminan más caracteres de las palabras que los lematizadores más débiles. Una forma es calcular el número medio de letras eliminadas cuando se aplica un lematizador a una colección de texto. Por lo tanto, suponga que las nueve palabras *"reacción"*, *"reaccionar"*, *"reaccionando"*, *"reaccionales"*, *"reaccionamos"*, *"reacciones"*, *"reacciona"* y *"reaccionemos"* se reducen todas a "*reacción*"; los números de caracteres eliminados son 0, 2, 4, 3, 4, 2, 1 y 4, respectivamente. Esto da una tasa de eliminación media de 20/8 = 2.5.

5. Distancia de Hamming (MHD): Esta distancia requiere dos cadenas de igual longitud y cuenta el número de posiciones correspondientes donde los caracteres son diferentes [29]. Por ejemplo:

La distancia de Hamming entre "**poder**" y "**goles**" es 3.

Si las cadenas son de diferentes longitudes, podemos utilizar la Distancia de Hamming modificada, MHD. Por lo tanto, supongamos que las longitudes de las cadenas son P y Q, donde P < Q, usamos la fórmula:

$$MHD = HD(1, P) + (Q-P) \quad (6)$$

donde HD (1, P) es la distancia de Hamming para los primeros caracteres P de ambas cadenas. Aplicando esto a una derivación, supongamos que la palabra "*fiestitas*" se convierte en "*fiesta*". En este caso, P = 6 y Q = 9, de modo que HD (1, P) = 1 por comparando "*fiesta*" con "*fiesti*", y (Q-P) = 3, dando MHD = 4. Claramente, es posible calcular el valor promedio de MHD para cada palabra en la muestra original.

5. Resultados experimentales

5.1. Diseño experimental

Una vez definido el contexto de la prueba, lo que incluyó el lote de palabras para la prueba y las métricas que se utilizarían para la comparación de los resultados conseguidos, se procedió a ejecutar cada versión del algoritmo:

- por un lado, el que se empleó para el SRI versión 2014 (en adelante SRI v2014) y
- por otro lado, la versión mejorada (en adelante SRI v2021).

En la versión original del algoritmo se encontró, comparando con el resultado de los lemas correctos, que no se estaba resolviendo ciertas terminaciones, tales como: *"ácea"*, *"acho"*, *"ucho"*, *"astro"*, *"astre"*, *"avo"*, *"bro"*, *"tión"*, *"zon"*, entre otras. Se modificó el código para incorporar estos sufijos faltantes en la nueva versión.

En la bibliografia consultada, se menciona que existen diferentes técnicas para realizar el proceso de lematización. En [30,31] se menciona lematizadores basados en diccionarios. Para el presente trabajo se agregó un nuevo paso, el mismo consiste en que una vez obtenido el lema resultante, se busca si el mismo corresponde o no a un verbo. De encontrarse, se reemplaza el verbo, por su correspondiente lema. En este proceso, en los 27.265 términos se encontraron 93 verbos. La lista con los 9.281 verbos agregada en el proceso se obtuvo del portal del Dr. Morales Flores. Esto permitió mejorar la versión que se encuentra en la web. Como se observar en la fig. 5.

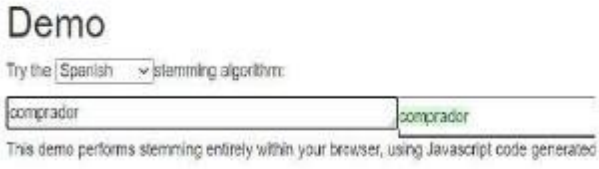


Figura 5. Devolución del lema del Snowball.

En la figura 6, se muestra la pantalla de salida del algoritmo modificado. Se observa los términos que resultan de cada paso, y devuelve correctamente la raíz de la palabra comprador.

Figura 6. Salida del lematizador modificado.

Una vez ejecutados los dos algoritmos y se obtuvieron los siguientes resultados:

A) Precisión de los algoritmos

La precisión, como se indicó en el apartado anterior, es un parámetro que se utiliza para evaluar la eficiencia del algoritmo de derivación. Cabe aclarar que, para esta experiencia, se consideran "*correctamente derivados*", cuando el lema devuelto por los algoritmos, coincide con el que se obtuvo del portal de Snowballstem.

En la tabla 1 se muestra los valores obtenidos para este paso.

Tabla 1. Comparación entre los dos algoritmos.

	Nº palabras correctamente derivadas	Porcentaje
SRI v2014	**17.280**	**63.38%**
SRI v2021	**24.267**	**89.00%**

Se puede observar, que las modificaciones planteadas en los párrafos anteriores, mejoraron, casi en un 26%, la cantidad de lemas que coinciden con los lemas del portal.

Luego, se procedió, a realizar los procesos para completar las distintas métricas presentadas en el apartado 4.1.

B) Rendimiento de los algoritmos.

Una vez concluidas las pruebas se realizó los cálculos de las distintas métricas presentadas en el apartado 4.1. En la tabla 2 y la figura 7 se observa como el algoritmo modificado obtiene en cada métrica, mejores valores que su antecesor.

Tabla 2. Tabla de rendimiento

Algoritmo	MWC	ICF	MHD	MCE	NPRD
SRI v2014	**1.66**	**0.40**	**1.88**	**1.88**	**23.732**
SRI v2021	**1.91**	**0.48**	**2.19**	**2.19**	**27.261**

Figura 7. Rendimiento de los algoritmos de lematización

Como se puede observar en la tabla 2 y en gráfico de la imagen 7, en todas métricas planteadas para, comparar el rendimiento de los algoritmos de lematización, dan mejores valores a la propuesta modificada.

5.2. Limitaciones

Los resultados presentan una buena precisión, pero estos dependen principalmente de disponer de un lemario completo de términos con sus respectivos lemas que estén correctamente verificados. Estas listas que deben ser preparadas exhaustivamente y necesitan muchas veces realizarse manualmente. Como se puede observar en la siguiente imagen, el algoritmo no resuelve correctamente dos términos como por ejemplo el caso de "*aconséjeme*" y "*aconseje*".

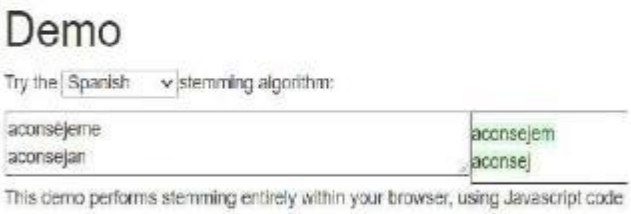


Figura 8. Ejemplo de lemas devuelto por el lematizador.

6. Conclusión y futuras líneas de trabajo

6.1 Conclusiones

En los últimos años hemos sido testigos de un creciente interés, por parte de distintas investigaciones, en obtener mejores SRI y, por optimizar los algoritmos de lematización. El propósito principal de esos algoritmos es reducir diferentes formas gramaticales como un sustantivo, adjetivo, verbo, adverbio, etc. a su forma raíz. Si bien, la mayor parte de los trabajos que se han realizado son para la lengua anglosajona, en este estudio se observó que los trabajos para el idioma español son escasos.

También se ha trabajado con una muestra reducida, mediante la evaluación de los algoritmos, se observa que la nueva versión proporciona una precisión del 89%, mejorando, en casi un 26%, a la versión anterior. Se concluye que es un aumento considerable, pero todavía resulta importante la cantidad de palabras mal derivadas. Este estudio abre la puerta a seguir profundizando en el tema.

concluye que o considerable, pero n, es un aumento considerable, todavía resulta importante la cantidad de palabras mal derivadas. Este estudio abre la puerta a seguir profundizando en el tema.

Se comprobó, además que, un enfoque basado en reglas no siempre da una salida correcta, respecto a las raíces generadas.

Por último, en lo que respecta al enfoque lingüístico, dado que estos métodos se basan en un léxico, es de suma importancia realizar un estudio exhaustivo del idioma español.

6.2 Futuras líneas de trabajo

A pesar de los avances realizados en este trabajo, en cuanto a mejorar un algoritmo de lematización para la lengua española, aún quedan diversos desafíos por resolver. Estos desafíos contemplan desde la confección de una lista de términos con sus respectivos lemas, hasta perfeccionar las etapas de identificación y tratamiento de todos los sufijos de la lengua. A continuación, se enumeran las diferentes líneas de trabajo futuro surgidas de estos desafíos:

- Modificar el orden de los pasos, propuesto en el algoritmo de Snowball, para optimizar los tiempos de procesamiento. Sobre todo, cuando se realicen las futuras pruebas con el lemario con más de un millón de palabras.
- Estudiar nuevos métodos de derivación. Como se comentó, existen distintas líneas de investigación, además de las mencionadas en este escrito.
- Profundizar en el estudio de la morfología léxica, ciencia que estudia la estructura de las palabras y las pautas que permiten formarlas o derivarlas a partir de otras.

Referencias

[1] Baeza-Yates, R. Y Ribeiro-Neto, B. (1999). Modern information retrieval. New York: Addison Wesley.

[2] Salton, G. Y Mcgill, M.J. (1983). Introduction to modern information retrieval. New York: McGraw-Hill.

[3] Martínez Méndez, F. Rodríguez Muñoz, J. Reflexiones sobre la evaluación de los Sistemas de Recuperación de Información: Necesidad, Utilidad y Viabilidad. Anales de documentación: Revista de biblioteconomía y documentación, ISSN 1575-2437, Nº. 7, 2004, pp 153-170. Disponible en: https://revistas.um.es/analesdoc/article/view/1651/1701. Último acceso: 20/07/2021.

[4] Blair, D. C. (1990). Language and representation in information retrieval. Ámsterdam: Elsevier.

[5] Zazo Rodríguez A. y otros. (2002). Recuperación de información utilizando el modelo vectorial. U. de Salamanca. Disponible en: http://eprints.rclis.org/13963/1/zazo2002 recuperacion.pdf. Último acceso: 20/07/2021.

[6] Salvador Olivan, José y Arquero Avilés, Rosario. (2006). Una aproximación al concepto de recuperación de información en el marco de la ciencia de la documentación. Investig. bibl [online]. 2006, vol.20, n.41, pp.13-43. ISSN 2448-8321. Último acceso: 20/07/2021.

[7] Martínez Méndez, F. (2004). Recuperación de información: modelos, sistemas y evaluación. Disponible en: http://eprints.rclis.org/16262/1/libro-ri.PDF. Último acceso: 20/07/2021.

[8] Figuerola C. y otros (2000) Diseño de un motor de recuperación de la información para uso experimental y educativo. Univ. de Salamanca. Disponible en: https://dialnet. unirioja.es/servlet/articulo?codigo=5555288. Último acceso: 20/07/2021.

[9] Herrero Pascual, Cristina. (2010). Manual de indización: teoría y práctica. Investigación bibliotecológica, 24(52), 239-240. http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2010000300010&lng=es&tlng=es. Último acceso: 20/07/2021.

[10] González, C. (2008). La recuperación de información en el siglo XX: Revisión y aplicación de aspectos de la lingüística cuantitativa y la modelización matemática de la información. UNLP. Disponible en: www.memoria.fahce.unlp.edu.ar/ tesis/te.350/te.350.pdf. Último acceso: 20/07/2021.

[11] Pickett V y Elson V. (1986). Introducción a la Morfología y Sintaxis. Disponible en: https://www.sil.org/system/files/reapdata/81/03/80/81038041802320181524090262276762975630/pickett_IntroMorfo_ed2.pdf. Último acceso: 20/07/2021.

[12] Gómez Díaz R. (2002). Estudio de la incidencia del conocimiento lingüístico en los sistemas de recuperación de la información para el español. I.S.B.N.: 84-7800-831-4 Ed. Universidad de Salamanca. Salamanca (España)

[13] Galvez, C.; Moya-Anegon, F.; Solana, V. H. Term conflation methods in information retrieval: non-linguistic and linguistic approaches. Journal of Documentation, v. 61, n. 4, p. 520-547,2005.

[14] Jurafsky D. & Martin J. (2020). Regular Expressions, Text Normalization, Edit Distance Speech and Language Processing. Disponible en: https://web.stanford.edu/~jurafsky/slp3/2.pdf. Último acceso: 20/07/2021.

[15] Mccabe A. (2020). Lemmatization and Stemming. A Brief Article on the History, Differences and Use-Cases of each Rooting Approach. Disponible en: https://medium.com/@alec.mccabe93/lemmatization-and-stemming-5b6b3718b49. Último acceso: 20/07/2021.

[16] Manning C.; Raghavan P. y Schütze H. An Introduction to Information Retrieval. Cambridge University Press. Cambridge, England. Online edition (c). 2009.https://nlp.stanford.edu/IR-book/pdf/irbookprint.pdf. Último acceso: 20/07/2021.

[17] Lorenzetti, Carlos M. (2011). Caracterización Formal y Análisis Empírico de Mecanismos Incrementales de Búsqueda basados en Contexto. Tesis Doctoral en Ciencias de la Computación - Universidad Nacional del Sur. Disponible en: https://arxiv.org/pdf/1810.04167.pdf. Último acceso: 20/07/2021.

[18] Hans P. Luhn. A statistical approach to mechanized encoding and searching of literary information. IBM Journal of Research and Development, 1(4):309-317, October 1957. Disponible en: http://openlib.org/home/krichel/courses/lis618 /readings/luhn57.pdf. Último acceso: 20/07/2021.

[19] Salton, G., Wong, A., & Yang, C. S. (1975). A Vector Space Model for Information Retrieval. Communications of the ACM, 18(11), 613–620. Disponible en: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.446.5101&rep=rep1&type=pdf. Último acceso: 20/07/2021.

[20] Tolosa G. y Bordignon F. (2008) Introducción a la Recuperación de Información. Conceptos, modelos y algoritmos básicos. Universidad Nacional de Luján, Argentina. Disponible en: http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf. Último acceso: 20/07/2021.

[21] Mendoza Olguín, G., Laureano de Jesús, Y., & Pérez de Celis Herrero, M. (2019). Métricas de similaridad y evaluación para sistemas de recomendación de filtrado colaborativo. Revista de Investigación en Tecnologías de la Información, 7(14), 224-240. Disponible en: https://www.riti.es/ojs2018/inicio/index.php/riti/article/view/175. Último acceso: 20/07/2021.

[22] Blanco E. y Sanz H. (2016). Algoritmos de clustering y aprendizaje automático aplicados a Twitter. Disponible en: https://upcommons.upc.edu/bitstream/handle/2117/82434/113257.pdf. Último acceso: 20/07/2021.

[23] Torres López, Carmen, & Arco García, Leticia. (2016). Representación textual en espacios vectoriales semánticos. Revista Cubana de Ciencias Informáticas, 10(2), 148-180. Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992016000200011&lng=es&tlng=es. Último acceso: 20/07/2021.

[24] Gómez S. A. (2001) Un Agente para clasificación y filtrado de páginas Web. Disponible en: https://core.ac.uk/download/pdf/296326901.pdf. Último acceso: 20/07/2021.

[25] Hohendahl A. Lematizador Morfosintáctico y Semántico Robusto con Flexionador y Estimador Idiomático, usando algoritmos eficientes y compactos para idiomas muy ricos en formas como el español. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/22789/Documento_completo.pdf?sequence=1. Último acceso: 20/07/2021.

[26] Bordignon F., W. Panessi. Procesamiento de variantes morfológicas en búsquedas de textos en castellano. Revista Interamericana de Bibliotecología, ISSN 0120-0976, Vol. 24, Nº. 1 (ENE-JUN), 2001, págs. 69-88. https://dialnet.unirioja.es/servlet/articulo?codigo=4291340. Último acceso: 20/07/2021.

[27] Gurusamy, Vairaprakash & Kannan, Subbu. (2017). Performance Analysis: Stemming Algorithm for the English Language. International Journal for Scientific Research and Development. 5. 2321-613. https://www.researchgate.net/publication/319525961_Performance_Analysis_Stemming_Algorithm_for_the_English_Language. Último acceso: 20/07/2021.

[28] jbossero@unlam.edu.ar o ej_moreno@unlam.edu.ar

[29] R.W. Hamming. (1950). Error Detecting and Error Correcting Codes. In The Bell System Technical Journal, vol 29, issue 2, American Telephone and Telegraph Company, USA, 1950.

[30] Olivas Varela J. (2011). Las técnicas de soft-computing en La recuperación de información. Disponible en: http://eventos.citius.usc.es/sematica2011/PDFs/traspas_JAngelOlivas.pdf. Último acceso: 20/07/2021.

[31] Jivani, Anjali. (2011). A Comparative Study of Stemming Algorithms. Int. J. Comp. Tech. Appl.. 2. 1930-1938.

Adecuación de un Sistema de Recuperación de Información para su utilización en un Contexto Jurídico

Osvaldo Sposito[1], Hugo Ryckeboer[1], Julio Bossero[1], Edgardo Moreno[1], Viviana Ledesma[1], Gastón Procopio[1], Lorena Matteo[1], Cecilia Gargano[1], Victoria Saizar[1], Patricio Macias[1], Juan Ojeda[1], Fabio Quintana[1], Laura Conti[2], Sergio García[3] y Gustavo Pérez Villar[4]

[1] Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigación Tecnológicas. Florencio Varela 1903. San Justo. La Matanza.

{sposito, hugor, jbossero, ej_moreno, vledesma, gprocopio, lmatteo, cgargano, vsaizar, pmacias, fquintana, jmojeda}@unlam.edu.ar

[2] Universidad Nacional de La Matanza. Departamento Derecho y Ciencia Política. lconti@unlam.edu.ar

[3] Palacio de Tribunales. Departamento Judicial de Morón. Alte. Brown. Piso 4. Morón. sergiogabriel.garcia@pjba.gov.ar

[4] Subsecretaría de Tecnología Informática del Poder Judicial de la Provincia de Buenos Aires. Palacio de Justicia, avenida 13 entre 47 y 48, primer piso (La Plata). Argentina. gperez@scba.gov.ar

RESUMEN

En las últimas décadas, las instituciones públicas, particularmente el Poder Judicial (PJ), con el desarrollo de las TICs, han generado un importante aumento en: la generación de documentos digitales, en los repositorios de los mismos y en los Sistemas de Recuperación de Información (SRI). Este trabajo se orienta a estudiar y proponer soluciones para la recuperación de documentos judiciales, se hace una propuesta para la construcción de la matriz de términos en un proceso de indización.

Palabras clave: SRI, Modelo Vectorial, Indización, Lematización.

CONTEXTO

La línea de investigación aquí presentada es parte del proyecto de investigación "*Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado*", perteneciente al programa de Investigaciones PROINCE (Programa de Incentivos para Docentes Investigadores) de la Secretaría de Políticas Universitarias del Ministerio de Educación de la Nación. Los integrantes del equipo son docentes e investigadores dependientes de las siguiente Unidades Académicas de la Universidad Nacional de La Matanza (UNLaM): el Departamento de Ingeniería e Investigaciones Tecnológicas (DIIT), y el Departamento Derecho y Ciencia Política, además, colaboran personal técnico de la Subsecretaría de Tecnología Informática del Poder Judicial de la Provincia de Buenos Aires.

1. INTRODUCCIÓN

En este trabajo se describe un proceso de indización, que consiste en extraer una serie de términos, representativos de los temas tratados en un documento, para utilizarlos después como puntos de acceso para la recuperación de esos documentos de un corpus jurídico. El propósito es brindar jurisprudencia similar a los profesionales del derecho luego de realizar una consulta. Entendiendo el concepto de jurisprudencia, como el conjunto de las sentencias de distintos fallos dictados por los tribunales de justicia u organismos judiciales de un Estado. En el campo del derecho, la jurisprudencia juega un papel importante como fuente del derecho; por ser la comprensión e interpretación de las normas jurídicas basada en las sentencias pasadas emitidas por órganos oficiales,

estas sustentan la aplicación de la ley en un caso concreto. En el PJ se producen una enorme cantidad de documentos jurídicos (dictámenes, expedientes, etc.) cada año, lo cual produce que esta fuente de derecho sea cada vez mayor, lo que impulsa a los profesionales del derecho a dedicar más tiempo a la búsqueda de una decisión relevante.

Basándonos en [1] coincidimos, en que los SRI están en continua mejoría, esto se debe a: la incorporación de utilidades dependientes de la expansión de su uso, el avance de las aplicaciones tecnológicas y el claro deslinde de sus funciones.

En [2], se referencia a Calvin N. Mooers como quien introdujo por primera vez en 1950 el término Recuperación de Información (en inglés Information Retrieval) en la literatura de documentación, la definió como *«la búsqueda de información en un stock de documentos, efectuada a partir de la especificación de un tema»*. Sólo un año más tarde, el mismo autor ampliaba esta definición al manifestar que la recuperación de información abarca los aspectos intelectuales de la descripción de información y su especificación para la búsqueda, y también cualquier sistema, técnica o máquina que se utilice para llevar a cabo la operación [3].

Según la bibliografía consultada [4-6], una SRI es un programa que interactúa entre un corpus y sus usuarios. Su efectividad depende del adecuado control del lenguaje de representación de los elementos de información y las búsquedas de sus usuarios. Para cumplir con sus objetivos, según Gabriel H. Tolosa y otros [5], un SRI debe realizar las siguientes tareas básicas:

- Representación lógica de los documentos y, opcionalmente, alma_ cenamiento del original.
- Representación de la necesidad de información del usuario en forma de consulta.
- Evaluación de los documentos respecto de una consulta para establecer la relevancia de cada uno.
- Ranking de los documentos considerados relevantes para formar el "conjunto solución o respuesta. Presentación de la respuesta al usuario.
- Retroalimentación de las consultas para aumentar la calidad de la respuesta.

Jaime Robredo en [5], asevera qué en cualquier área del conocimiento, los términos con significado se pueden utilizar como descriptores para representar el contenido de documentos escritos, en los procesos de indización y organización de la información, así como para formular preguntas en el proceso de recuperación de información. Tolosa en [5] afirma que el proceso se puede dividir en las siguientes etapas:

- Análisis lexicográfico: Se extraen las palabras y se normalizan.
- Reducción (Tokenización) de palabras vacías o de alta frecuencia.
- Lematización: Se reducen palabras morfológicamente parecidas a una forma base o raíz, con la finalidad de aumentar la eficiencia de un SRI.
- Asignación de pesos o ponderación de los términos que componen los índices de cada documento.

Los SRI implementan una gama diversa de estructuras de datos, algoritmos y técnicas de recuperación de información, por ello, se precisa de un modelo conceptual donde se determinen: el tipo de almacenamiento, operaciones sobre los términos, modelos de búsqueda con base patrones exactos o los modelos inexactos los cuales contendrán las técnicas probabilísticas, los modelos lógicos y los espacios vectoriales [8]. En el trabajo de Martínez Méndez, se puede encontrar un estudio detallado de los distintos modelos de RI existentes. Uno de los modelos más utilizados [4-5], es el Modelo de Espacio

Vectorial. En este modelo, el texto es representado por un vector de términos, los términos comúnmente son palabras; cualquier texto puede ser representado por un vector en un espacio dimensional Salton en el año 1975 [9]. En un espacio de documento que consiste en documentos D_i, cada uno identificado por uno o más términos de índice T_j; los términos pueden ser ponderados de acuerdo a su importancia, o no ponderados con pesos restringidos a 0 y 1. En el modelo los documentos se representan a partir de vectores, de la siguiente manera:

$$D_i = (T_1, T_2, ..., T_j) \quad (1)$$

En la Figura 1 se muestra un espacio de índice tridimensional, donde cada elemento se identifica con hasta tres términos distintos.

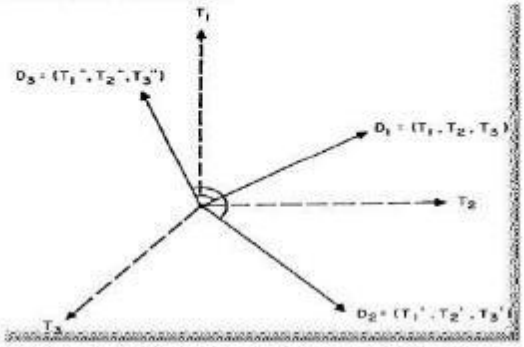


Figura. 1. Representación vectorial espacial de los documentos [9].

Una consulta se puede ver como un documento por lo tanto se puede ver como un vector.

Matemáticamente, una de formas de establecer la cercanía de dos vectores es calcular el coseno del ángulo que forman los dos vectores entre sí. Esta fórmula tiene la ventaja de su bajo esfuerzo computacional y es independiente de los módulos de los vectores. De manera similar, se puede calcular el coseno del ángulo entre cada vector de documento y el vector de consulta para encontrar su cercanía. Para encontrar un documento relevante para el término de la consulta, se calcula la puntuación de similitud entre cada vector del documento y el vector del término de la consulta aplicando la similitud del coseno. Finalmente, aquellos documentos con puntajes de similitud altos se considerarán documentos relevantes para la consulta. [9].

Como se comentó, dentro de la indización se encuentra la lematización, que es una técnica empleada en la recuperación de datos en los SRI, que sirve para reducir variantes morfológicas de la forma de una palabra a raíces comunes o lexemas; con el fin de mejorar la habilidad de los motores de búsqueda y, a consecuencia, los resultados de las consultas. Básicamente, este consiste en remover el plural, el tiempo, o los atributos finales de la palabra [5,6,10]. En el trabajo de González [6], afirma que "*cuando se realiza la extracción de palabras de un texto se obtiene una gran cantidad de entradas con formas verbales conjugadas y variantes de concordancia. Logrando la reducción morfológica de todas estas variantes se busca que el usuario recupere tanto los textos que contienen sus términos de búsqueda, como aquellos que contienen las formas derivadas de esos términos…*". Cabe aclarar que, en este proyecto, nosotros también simplificamos las apariciones de sustantivos y adjetivos. Los algoritmos de lematización más conocidos son: Lovins[1] (1968), Porter[2] (1980) y Paice[3] (1990). Originalmente todos fueron hechos para el inglés, y se diferencian en la eficiencia

1 http://snowball.tartarus.org/algorithms/lovins/stemmer.html

2 https://tartarus.org/martin/PorterStemmer/

3 https://www.scientificpsychic.com/paice/paice.html

del código y la elección de sufijos que identifican y eliminan. Una modificación del algoritmo trabajo de Porter, es el algoritmo de Snowball[4]. Este puede mapear palabras que no están en inglés. Estos algoritmos permiten realizar "*derivaciones*", esto es remover los sufijos comunes morfológicos e inflexionales de palabras literalmente diferentes, pero con una "*raíz*" común, que pueden ser consideradas como un sólo término. Este algoritmo requiere de un conjunto de pasos para llegar a la raíz.

2. LÍNEAS de INVESTIGACIÓN y DESARROLLO

El presente trabajo tiene como eje central el desarrollo de un SRI. Entre las líneas de investigación a considerar en este proyecto se pueden mencionar:

- El problema de la recuperación de información, el modelo vectorial y la forma de almacenar los términos de una colección (corpus) de pruebas.
- La paralelización del proceso de Indexación Semántica Latente (ISL). Se estudian las librerías: Compute Unified Device Architecture (CUDA) y CUDA Basic Linear Algebra Subprograms (CuBlas), aplicadas a una arquitectura híbrida.
- La aplicando el patrón de arquitectura Modelo-Vista-Controlador (MVC), para desarrollos WEB. Aplicando el lenguaje de programación C#.
- Estudio de la librería REGEX., para resolver las Expresiones Regulares (ER).
- Estudio y evaluación de distintos algoritmos de ranking para Documentos. Las pruebas serán realizadas tomando como base un corpus jurídico real.

3. RESULTADOS OBTENIDOS/ESPERADOS

Durante el año 2021 se ha trabajado, principalmente, en dos temas, por un lado, en el estudio, análisis y modificación de algoritmos y técnicas que permitan la lematización de términos, y por otro, en el proceso que permita incorporar, de un corpus jurídico, las fechas y las referencias de la norma jurídica actual, mediante el Reconocimiento de Entidades Nombradas (tales como Acordadas, Artículos, Leyes, entre otros), que componen los distintos textos judiciales, utilizando Expresiones Regulares (ER).

Se presentaron en distintos congresos las siguientes publicaciones:

1. "***Propuesta para la construcción de un corpus jurídico utilizando Expresiones Regulares***". Presentado en el XXVII Congreso Argentino de Ciencias de la Computación (CACIC). Salta. Argentina [8].

Una ER es una notación algebraica para caracterizar un conjunto de cadenas [11]. Son particularmente útiles para la búsqueda en textos, cuando se tiene un patrón y un corpus de textos donde buscar. En este trabajo se demostró que es posible incorporar en el proceso de Análisis lexicográfico Expresiones Regulares para incorporar fechas y Entidades Nombradas a una matriz de términos. Dentro de las tareas a desarrollar, durante este año, se puede mencionar:

- Incorporar la codificación propuesta al SRI implementado por el proyecto PROINCE mencionado en la introducción.
- Analizar otros algoritmos y técnicas de derivación.
- Estudiar otras librerías existentes de ER.
- Realizar una clasificación de todas las EN dentro de la norma jurídica Argentina.

2. "***Implementación de un lematizador***

[4] https://snowballstem.org/demo.html

para la lengua española". Trabajo presentado en el Workshop del IX Congreso Nacional de Ingeniería en Informática/Sistemas de Información. CONAIISI 2021. Mendoza. Argentina.

En este trabajo se muestra una modificación realizada al algoritmo de Snowball. Mejorando en un 26% la lematización de términos. Se prevé para este año:

- Modificar el orden de los pasos, propuesto en el algoritmo de Snowball, para mejorar los tiempos de procesamiento.
- Estudiar nuevos métodos de derivación.
- Profundizar en el estudio de la morfología léxica, ciencia que estudia la estructura de las palabras y las pautas que permiten formarlas o derivarlas a partir de otras.

4. FORMACIÓN DE RECURSOS HUMANOS

La presente línea de investigación la lleva adelante un equipo de 15 integrantes provenientes de dos departamentos de la UNLaM, el DIIT y el Departamento de Derecho y Ciencia Política.

- 1 alumno de grado. En el año 2021 se gradúo en la carrera de Ingeniería de Informática.
- 2 asesores especialistas externos. (uno perteneciente al Poder Judicial de la Provincia de Buenos Aires y un Secretario de Juzgado).

5. BIBLIOGRAFÍA

[1] Galindo Ayuda, F. (2020). Avances en sistemas jurídicos de recuperación de documentos. Scire: Representación Y organización Del Conocimiento, 26(1), 63–74. https://doi.org/10.54886/scire.v26i1.4698. Fecha de consulta: 07/02/22

[2] S. Oliván, J.A., & Arquero Avilés, Rosario. (2006). Una aproximación al concepto de recuperación de información en el marco de la ciencia de la documentación. Investigación bibliotecológica, 20(41), 13-43. Disponible en: http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2006000200002&lng=es&tlng=es. F. de consulta: 07/02/22

[3] C.N. Mooers, "The theory of digital handling of non–numerical information and its implications to machine economics", en Technical Bulletin No. 48. Cambridge, MA: Zator Co., 1950 (Ponencia presentada en Association for Computing Machinery, Rutgers Univ., New Brunswick, NJ, 1950, March 29).

[4] Kuna, H., Rey, M., Martini, E., Solonezen, L. & Podkowa, L. Desarrollo de un Sistema de Recuperación de Información para Publicaciones Científicas del Área de Ciencias de la Computación. Rev. Latinoamericana de Ingeniería de Software, (2014). 2(2): 107-114. http://revistas.unla.edu.ar/software/article/view/81. Fecha de consulta: 07/02/22

[5] Tolosa G. & Bordignon, F. Introducción a la Recuperación de Información: Conceptos, modelos y algoritmos básicos. UNdeL, Argentina, (2008). En línea: http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf. Fecha de consulta: 07/02/22

[6] González, C. M. La recuperación de información en el siglo XX. Revisión y aplicación de aspectos de la lingüística cuantitativa y la modeliza-ción matemática de la información UNLP. (2008) Disponible en: https://memoria.fahce.unlp.edu.ar/tesis/te.350/te.350.pdf. Fecha de consulta: 07/02/22

[7] Robredo, J. (2019). Otimização dos processos de indexação dos documentos e de recuperação da informação mediante o uso de instrumentos de controle terminológico. Ciência Da Informação, 47(1). Recuperado de http://revista.ibict.br/ciinf/article/view/4431. Fecha de consulta: 07/02/22.

[8] Martínez Méndez, F. (2004). Recuperación de información: modelos, sistemas y evaluación. Disponible en: http://eprints.rclis.org/16262/1/libro-ri.PDF. Fecha de consulta: 07/02/22.

[9] Salton, G., Wong, A., & Yang, C. S. (1975). A Vector Space Model for Information Retrieval. Communications of the ACM, 18(11), 613–620. Disponible en: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.446.5101&rep=rep1&type=pdf. F. consulta: 07/02/22.

[10] Zazo Rodríguez A. y otros. (2002). Recuperación de información utilizando el modelo vectorial. U. de Salamanca. Disponible en: http://eprints.rclis.org/13963/1/zazo2002recuperacion.pdf. Fecha de consulta: 07/02/22.

[11] Robaldo, L. y otros. Compiling regular expressions to extract legal modifications. 250. 133-141. 10.3233/978-1-61499-167-0-133. (2012).

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

Resultados preliminares de una técnica de localización de documentos en espacios métricos utilizando K-means

Osvaldo Sposito[1], Julio Bossero[1], Edgardo Moreno[1], Viviana Ledesma[1], Gastón Procopio[1], Lorena Matteo[1], Cecilia Gargano[1], Victoria Saizar[1], Patricio Macias[1], Juan Ojeda[1], Fabio Quintana[1], Laura Conti[2], Sergio García[3] y Gustavo Pérez Villar[4]

[1] Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigación Tecnológicas. Florencio Varela 1903. San Justo. La Matanza.
{sposito, jbossero, ej_moreno, vledesma, gprocopio, lmatteo, cgargano, vsaizar, pmacias, jmojeda}@unlam.edu.ar

[2] Universidad Nacional de La Matanza. Departamento Derecho y Ciencia Política. lconti@unlam.edu.ar

[3] Palacio de Tribunales. Departamento Judicial de Morón. Alte. Brown. Piso 4. Morón. sergiogabriel.garcia@pjba.gov.ar

[4] Subsecretaría de Tecnología Informática del Poder Judicial de la Provincia de Buenos Aires. Palacio de Justicia, avenida 13 entre 47 y 48, primer piso (La Plata). Argentina. gperez@scba.gov.ar

Resumen

En un modelo basado en espacio vectorial, un Sistema de Recuperación de Información, para encontrar los documentos más similares a una consulta, debe enfrentar a esta, con todos los documentos existentes en el corpus. En este enfoque, todos los objetos deben ser convertidos a vectores de características numéricas y la comparación entre ellos se debe realizar calculando una métrica de proximidad o similitud. Este proceso, si bien es el mejor, requiere un alto costo computacional. Varios estudios analizan la posibilidad de fraccionar el corpus de modo tal de reducir el tiempo, sin perder calidad de la primera respuesta que entrega el sistema frente a una consulta. Se propone que los segmentos contengan documentos afines con la finalidad que muchas consultas queden resueltas por examen de un solo segmento, aunque esa respuesta adolezca de algunos documentos. Para llevar a cabo este segmentación, se emplean técnicas de agrupación automático, que permiten organizar, extraer características y clasificar objetos según su similitud o proximidad. Este artículo presenta el desarrollo de un programa parametrizable, que permite la construcción de un espacio métrico de n cantidad de documentos, con diversos tamaños de dimensionalidad y utilizando distintas normas (distancia). Esta herramienta permite a los usuarios visualizar rápidamente los resultados de la búsqueda, utilizando diferentes configuraciones.

1. Introducción

Este grupo de investigación perteneciente a la Universidad Nacional de La Matanza (UNLaM), desde el año 2013 viene investigando sobre distintos procesos que conforman un Sistema de Recuperación de Información (SRI) [1-7]. En el año 2021, se presentó a través del Programa de Incentivos para Docentes Investigadores de la Secretaría de Políticas Universitarias (PROINCE), un proyecto con el título *"Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado"*. En este proyecto se propone una versión renovada de un SRI, orientado de modo especial a la recuperación de documentos jurídicos (dictámenes, expedientes, etc.). Uno de los objetivos es la construccion corpus jurídico lo suficientemente voluminoso para poder evaluar los tiempos de respuesta del SRI y la pertinencia de los documentos recuperados.

El objetivo de todo sistema de recuperación de documentos es sugerir una lista de documentos ordenados de acuerdo a la cercanía que manifiestan al requerimiento formulado. Sólo el examen de los documentos por parte del requeridor confirma el mayor o menor acierto del sistema [1]. Cualquier usuario de buscadores de la WEB tiene la experiencia de que no siempre la primera respuesta que ofrece el sistema sea la más adecuada. Es lógico pensar que, conforme los corpus aumentan de tamaño, ese tiempo aumenta por ser proporcional al tamaño. Para reducir ese tiempo se puede recurrir a un aumento de potencia de cómputo los que encaran ese camino recurren especialmente al paralelismo o a un pre-proceso que tienda a acelerar todas las consultas posteriores o al menos a un alto porcentaje de las mismas. En el trabajo presentado en [3], se expuso que esto requiere dos algoritmos preparatorios:

a) uno que particione el corpus utilizando una noción de vecindad o similitud y

b) el entrenamiento de un algoritmo de clasificación que direccione la consulta hacia la parte más promisoria.

posteriormente, por cada consulta, se debe ejecutar dos pasos:

1) Aplicar el algoritmo que direcciona la consulta hacia una de las partes.

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

2) Enfrentar la consulta con cada documento de esa parte para determinar su grado de adecuación y posterior posición en la lista de documentos sugeridos.

Si la parte elegida efectivamente contiene un alto porcentaje de los documentos que hubieran encabezado la lista de haber hecho el proceso sobre la totalidad el usuario no sentiría demasiado la baja de la exhaustividad[1]. Evidentemente, habrá consultas cuya respuesta completa esté repartida entre varias partes, pero mientras haya en la página inicial suficientes documentos representativos de la respuesta ideal para que el usuario los examine hay tiempo de procesar otras partes y mostrarle cuando solicite la segunda página lo que hubiera faltado en la primera [3]. En este trabajo se ensaya el camino inverso, en lugar de efectuar macro-agrupamientos se comienza con micro agrupamientos. A igual que en el ensayo de la técnica citada al carecer de una voluminosa cantidad de documentos para experimentar el proceso de recuperación, en el presente trabajo se construye artificialmente vectores que representan documentos y consultas en lenguaje natural. Estos vectores, conformarán un corpus de la experimentación. El mismo, utilizando una técnica de clasificación no supervisada (clusterización), K-means [8], agrupará los objetos en *k* grupos basándose en sus similitudes o cercanías. Con estas simulaciones se pretende demostrar que hay un modo de estructurar los datos, por una única vez, tal que sugerir los mejores documentos no requiere examinarlos todos.

Como es sabido, el modelado y la simulación es cada vez más importante para el análisis y diseño de sistemas complejos [9]. El objetivo de estas técnicas es poder ayudar o dar el soporte necesario a distintos interesados durante el proceso de diseño, análisis y diagnosis de sistemas ingenieriles.

En el punto 2 se expone el marco teórico de las distintas tecnologías que abarcan este escrito, en el punto 3 se hace un pequeño resumen de algunos trabajos similares al presentado en este trabajo, en el punto 4 se explica la herramienta presentada, junto con la descripción de cada una de las salidas, luego en el punto 5 se explica otra aplicación que sirve para analizar los resultados obtenidos, y por último en el punto 6 se exponen las propuestas a trabajos futuros.

2. Marco Teórico

En este apartado se introducen algunos conceptos básicos sobre los temas que abarca este trabajo. Se describen brevemente cada uno de ellos.

2.1 Sistema de Recuperación de Información

Un SRI puede definirse como la representación, almacenamiento, organización y el acceso a elementos de información [1]. Estos sistemas permiten la recuperación de los documentos, previamente almacenados, por medio de consultas. Salton piensa que "*cualquier SRI puede ser descrito como un conjunto de ítems de información (Documentos), un conjunto de peticiones (Querys) y algún mecanismo (Procesador) que determine qué ítems satisfacen las necesidades de información expresadas por el usuario en la petición*" [9].

Los autores Gabriel Tolosa y Fernando Bordignon [10], dan una clasificación de los documentos en estructurados y no estructurados tal como se puede visualizar en la Figura 1. Los primeros son aquellos en los que se pueden reconocer elementos estructurales con una semántica bien definida, mientras que los segundos corresponden a texto libre, sin formato. De acuerdo a esta clasificación se pude diferenciar distintos modelos de recuperación de información, de acuerdo a características estructurales de los documentos.

Figura 1. Modelos de SRI. [10]

En este trabajo, sólo se hace referencia al modelo vectorial.

2.2 El Modelo Vectorial

El modelo vectorial de recuperación de información fue presentado por Gerard Salton en 1975 y posteriormente asentado en 1983 junto con McGill [11-12]. Propone un marco en el que es posible el emparejamiento parcial a diferencia del modelo de recuperación booleano, asignando pesos no binarios a los términos índice de las preguntas y de los documentos. Estos pesos de los términos se usan para computar el grado de similitud entre cada documento guardado en el sistema y la pregunta del usuario [4][5].

El modelo vectorial se basa en el grado de similaridad de una consulta dada por el usuario con respecto a los documentos de la colección cuyos términos fueron ponderados mediante TF-IDF (del inglés Term frequency – Inverse document frequency), frecuencia de término – frecuencia inversa de documento (o sea, la frecuencia de ocurrencia del término en la colección de documentos), es una medida numérica que expresa cuán relevante es una palabra para un documento en un corpus [11-13]. Esta medida se utiliza a menudo como un factor de ponderación en los SRI. La forma más natural de describir un documento es a través de los términos que lo componen, mientras más palabras se utilicen, más oportunidades habrá de recuperar un documento. Entonces, esta medida, pondera el uso de una determinada palabra dentro de un conjunto de documentos. Supone que dependiendo de dicho valor un elemento es importante y relevante para la clasificación de documentos frente a la consulta de un usuario. Este indicador se obtiene con el producto entre TF y IDF (TF x

[1] Porcentaje de documentos relevantes recuperados, sobre el total de documentos relevantes que hay.

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

IDF). A continuación, se busca clarificar la esencia de ambos conceptos:

- TF: *Frecuencia de términos*, es la cantidad de veces que aparece dicho término en dicho documento. TF puede calcularse de manera «sencilla» como el número de veces que se repite un término en un documento o de formas mucho más complejas con expresiones matemáticas como operadores booleanos o logaritmos.
  TF = Nº Total del término en el documento / Nº Total de palabras en el documento.

- IDF: *Frecuencia inversa de documento*, se utiliza para disminuir el peso de aquellos términos que son muy frecuentes en varios textos que se están considerando. Igualmente, en este caso puede expresarse matemáticamente con expresiones que incluyen logaritmos o de manera simplificada:
  IDF = Nº Total de documentos / Nº de documentos con el término buscado.

Ejemplo de cálculo del TF*IDF:

Suponiendo que se está leyendo un documento de 100 palabras dónde la palabra «río» aparece 3 veces.

El TF se calcularía: TF = 3/100 = 0,03

Se encuentran 10 millones de documentos y donde la palabra «río» supongamos aparece en 1.000.

El IDF se calcularía: $\log_2 (10.000.000/1.000) = 4$

Finalmente al aplicar la expresión completa del TF-IDF se obtiene: TF x IDF = 0.03 x 4 = 0,12

Resumiendo, esto sugiere que un factor que indique la frecuencia de un término debe estar presente en una métrica de documentos o de consultas. El valor TF-IDF aumenta proporcionalmente al número de veces que una palabra aparece en el documento, pero es compensada por la frecuencia de la palabra en la colección de documentos, lo que permite manejar el hecho de que algunas palabras son generalmente más comunes que otras [13].

La idea básica del modelo vectorial reside en la construcción de una matriz (comunmente llamada matriz de términos/documentos), donde las filas fueran estos últimos y las columnas correspondieran a los términos incluidos en ellos. Las filas de esta matriz (que en términos algebraicos se denominan vectores) serían equivalentes a los documentos que se expresarían en función de las apariciones (frecuencia) de cada término. De esta manera, un documento podría expresarse de la siguiente manera: d_1=(1, 2, 0, 0, 0, ..., 1, 3): Siendo cada uno de estos valores el número de veces que aparece cada término en el documento. La longitud del vector de documentos sería igual al total de términos de la matriz (el número de columnas). De esta manera, un conjunto de *m* documentos se almacenaría en una matriz de *m* filas por *n* columnas, siendo *n* el total de términos almacenados en ese conjunto de documentos [12].

Para definir la distancia entre dos vectores en R^n, se utilizará R^2 como modelo. La fórmula de la distancia en geometría analítica nos dice que la distancia d entre dos puntos en el plano, (u_1, u_2) y (v_1, v_2), es: [13]

$$d = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2}$$

En terminología vectorial, esta distancia puede considerarse como la longitud de u–v, donde $\mathbf{u} = (u_1, u_2)$ y $\mathbf{v} = (v_1, v_2)$, como se observa en la Figura 2. Es decir,

$$\|u - v\| = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2}$$

lo cual conduce a la siguiente definición.

$$d(u, v) = \|u - v\| = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2}$$

Esta función, aplicada a 3 o más dimensiones, es una generalización del teorema de Pitágoras y se denomina distancia euclidiana.

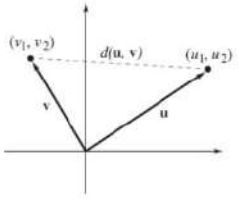


Figura 2. Distancia entre dos vectores [13]

Volviendo a los SRI, en el modelo vectorial, los pesos de los términos se emplean para calcular la similitud entre los documentos y las consultas [12]. Este modelo ordena los documentos recuperados de manera decreciente de acuerdo, al grado de similitud. Los documentos y consultas son representados como vectores de dimensión t (número total de términos en la colección) y una forma de calcular la similitud es con el 1-coseno del ángulo entre el vector que representa a la consulta y el que representa al documento. Figura 3. En la imagen que pueden observar los vectores d_1, d_2 y q, que representan a los documentos 1, 2 y a la consulta respectivamente. Estos vectores son números que tienen una magnitud y una dirección. Tanto la magnitud como la dirección deben medirse con respecto al espacio en el que se define el vector. Cada dimensión del espacio representa una característica de interés y un vector representa la medida en que el objeto del modelo tiene esas características. Por tanto, un vector es una lista de números: uno para cada característica que forma parte del espacio modelo. La dirección del vector es la que va desde el origen del espacio hasta el punto definido por esos números.

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

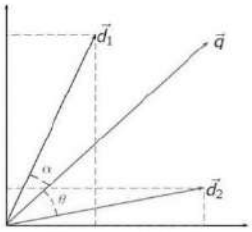


Figura 3. Representación de objetos en el modelo vectorial [12]

Dentro de esta clasificación, como modelo alternativo, está el modelo de Indexación Semántica Latente (Latent Semantic Indexing, LSI) [4] que descansa sobre la Descomposición de Valores Singulares (en inglés Singular Value Descomposition, SVD) [5] que es una técnica de factorización de matrices que permite descomponer una matriz. La SVD permite construir una nueva matriz de menor rango que la original y que entre todas ellas sea la de menor distancia euclídea de la matriz original. Esta reducción de rango permite disminuir la dimensión de los vectores representativos de los documentos. Como se verá más adelante la reducción de dimensión en la representación de documentos es esencial para la efectividad de la técnica propuesta.

Con la LSI se pretende la resolución de perturbaciones en la recuperación de información debido a problemas de sinonimia y polisemia o equivocidad del habla corriente. Por ejemplo, si se desea buscar la palabra "estación", la cual tiene múltiples significados (polisemia) una búsqueda literal de la palabra produciría muchos resultados posibles (estación de tren, estación del año, etc.). Si lo que se desea buscar es "estación del año", podrían interesar resultados de palabras distintas, pero con un significado igual o similar, por ejemplo "temporada", "época" y así por el estilo (sinonimia). La LSI permite la búsqueda por conceptos o definiciones (en contraposición a la búsqueda literal). En este punto, interesa aclarar al lector, acerca del término "*orientaciones temáticas*", que es utilizado para hacer referencia a la cantidad de características numéricas del vector. Resumiendo, ese número representa que, palabras que se utilizan en un mismo contexto tiene significados similares en un SRI y esto permite identificar patrones entre los términos contenidos en una colección de textos.

En [14], los autores señalan que, son varias las investigaciones, donde se aplica la búsqueda por similitud o por proximidad para recuperar dentro de bases de datos métricas, objetos que resulten parecidos o relevantes a una determinada consulta. Este proceso se modela matemáticamente a través de un espacio métrico, en el cual los objetos son representados como una caja negra donde la única información disponible es la distancia de este objeto a los otros.

2.3 Espacios Métricos: Generalidades

En [15], se encuentra una definición dada por el matemático francés Maurice Fréchet: un espacio métrico es un par (X, d), donde X es un conjunto arbitrario no vacío y $d : X \times X \rightarrow \mathbb{R}_{\geq 0}$ una aplicación, llamada distancia o métrica, tal que, para cualesquiera $x, y, z \in X$, se verifica:

$$(1)\ d(x, y) \geq 0. \quad \textbf{(1)}$$

$$(2)\ d(x, y)=0 \Longleftrightarrow x = y. \quad \textbf{(2)}$$

$$(3)\ d(x, y) = d(y, x). \quad \textbf{(3)}$$

$$(4)\ d(x, z) \leq d(x, y) + d(y, z) \quad \textbf{(4)}$$

Los métodos tradicionales de búsqueda están orientados al tratamiento de datos con una estructura determinada, generalmente mediante atributos que reflejan las propiedades de los objetos y son representados mediante registros almacenados en la base de datos. Las búsquedas en este contexto, donde los atributos son generalmente de tipo estructurado (numérico, alfanumérico, de tiempo, entre otros), están basadas en la búsqueda exacta y sus variantes de búsqueda por rango. Ésta última se realiza entre valores determinados, restringiéndose la igualdad a unas posiciones determinadas, por ejemplo, consultar por una cadena alfanumérica con algunos caracteres específicos [15].

Los objetos multimedia utilizados en las aplicaciones actuales son tipos de datos no estructurados, éstos pueden ser imágenes, videos, audio, documentos de texto, secuencias de ADN, entre otros [16]. Generalmente no poseen una estructura determinada, razón por la cual una búsqueda total exacta es inaplicable. Los datos multimedia presentan una gran dificultad para reproducirse en las mismas condiciones (por ejemplo: dos imágenes tomadas de una misma escena casi nunca son exactamente iguales), siendo muy sensibles a situaciones de ruido, distorsiones, entre otras degradaciones. Por estos motivos, si se desea buscar un objeto multimedia en una base de datos, se debe realizar a través de una búsqueda por similitud, es decir determinar cuáles son los elementos más semejantes, parecidos o "cercanos" a algún otro, definido como objeto de consulta [16].

Por lo expuesto hasta el momento, se puede decir que, los espacios métricos pueden ser de dimensión infinita. Una base de datos métrica es realmente un espacio vectorial de dimensión finita D, donde los elementos son vectores de coordenadas reales, es decir elementos de $\mathbb{R}^D$. Alrededor de este modelo de datos se han realizado muchos desarrollos, tanto de índices y operaciones, como de lenguajes de consulta. Sin embargo, generalmente estos desarrollos no pueden ser extendidos a las bases de datos métricas, donde sólo se dispone de la información de distancia entre los objetos [16].

2.4 Espacios normados

Una norma en un espacio vectorial X es una aplicación $\|\cdot\|: X \rightarrow \mathbb{R}$, que a cada vector $x \in X$ hace corresponder un número real no negativo $\|x\|$, verificando las tres condiciones siguientes:

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

(N.1) Desigualdad triangular:

$$|x+y| \leq |x|+|y| \; \forall x,y \in X \quad (5)$$

(N.2) Homogeneidad por homotecias:

$$|\lambda.x|=|\lambda| \; |x| \; \forall x \in X, \forall \lambda \in R \quad (6)$$

(N.3) No degeneración:

$$x \in X, |x| = 0, \text{ si y sólo si } x = 0 \quad (7)$$

Un espacio normado es un espacio vectorial X, en el que se ha fijado una norma |·|. Para cada x ∈ X, se dice también que el número real |x| es la norma del *vector* x. En este trabajo, se utilizarán las normas 1, 2 y 4. A continuación se expresan sus respectivas fórmulas:

La noma /1 viene dada por

$$\|X\|_1 = \sum_{i=1}^{n} |X_i|$$

La noma /2 o norma Euclidiana es dada por:

$$\|X\|_2 = \left(\sum_{i=1}^{n} X_i^2 \right)^{1/2}$$

Por último, la norma 4.

$$\|X\|_4 = \left(\sum_{i=1}^{n} X_i^4 \right)^{1/4}$$

2.5 Introducción al Aprendizaje Automático

El aprendizaje automático o aprendizaje de máquinas (del inglés, machine learning ML), es una rama de la Inteligencia Artificial (IA), cuyo objetivo es desarrollar sistemas que aprenden, o mejoran el rendimiento, en función de los datos que consumen. Estas técnicas permite hacer predicciones, frecuentemente precisas a partir de observaciones con datos previos [8]. Podemos considerar tres tipos de aprendizajes:

- Aprendizaje supervisado: Se dispone de un conjunto de datos (llamados de entrenamiento) y cada dato está asociado a una etiqueta o clase. Se construye un modelo que en una la fase de entrenamiento (training) utiliza dichas etiquetas, para determinar si el dato está clasificado correcta o incorrectamente por el modelo. La idea es que se utilice el modelo para clasificar nuevos datos que no cuenten con etiqueta.
- Aprendizaje no supervisado: los datos no tienen etiquetas (o no queremos utilizarlas) y estos se clasifican a partir de su estructura interna de los mismos (propiedades, características).
- Aprendizaje por refuerzo: algunos datos de entrenamiento tienen etiquetas, pero no todos.

Este trabajo hace referencia solamente al algoritmo de clasificación no supervisado K-means.

El algoritmo K-means o K-medias es un método de agrupamiento que divide un conjunto de n observaciones en k grupos distintos gracias a valores medios. Como ya se comentó, pertenece al ámbito de los algoritmos No Supervisados, ya que las n observaciones no cuentan con un atributo clase o etiqueta, que nos diga de qué grupo es cada dato, siendo los datos agrupados según sus propiedades o características [17].

El agrupamiento de las n observaciones en los k grupos distintos se realiza minimizando la suma de distancias entre cada observación y una ubicación real o imaginaria que representa el centro del grupo. Cada punto de datos se asigna a cada uno de los grupos mediante la reducción de la suma de cuadrados en el grupo. A esta ubicación se la denomina centroide. La distancia más común es la distancia euclídea. El algoritmo cuenta con tres pasos:

- Inicialización: una vez escogido k (número de grupos), se establecen los centroides en el espacio de los datos, por ejemplo asignando los k puntos aleatoriamente.
- Asignación de las observaciones a los centroides: cada observación es asignada al centroide más cercano a ella usando la medida de distancia que se determine.
- Actualización de los centroides: se actualiza la posición de los centroides de cada grupo tomando como posición la media de la localización de las observaciones de dicho grupo.

Se repiten los pasos 2 y 3 hasta que los centroides se quedan fijos, o se mueven por debajo de una distancia umbral fijada [18]. En la siguiente figura se explica el funcionamiento del algoritmo, la imagen fue obtenida de [18], los ejemplos de entrenamiento se muestran como puntos y los centroides de clúster se muestran como cruces.

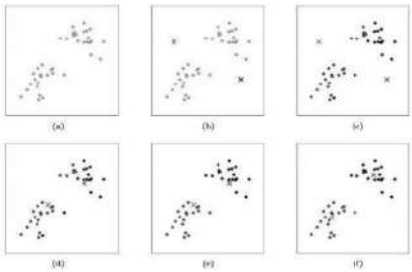


Figura 4. Secuencia del algoritmo K-means [18]

En estea Figura se observa, en un espacio bidimensional, las siguientes secuencias: (a) el conjunto de datos original; (b) centroides de grupos iniciales aleatorios; (c-f) ilustración de ejecutar dos iteraciones de -Means. En cada iteración, se asignó cada ejemplo de entrenamiento al centroide de clúster más cercano (que se muestra al "pintar" los ejemplos de entrenamiento del mismo color que el centroide de clúster al que está asignado); luego se mueve cada centroide de grupo a la media de los puntos asignados a él.

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

3. Trabajos relacionados

Existen varios trabajos que siguen la idea de segmentar o agrupar, utilizando alguna técnica de IA, un repositorio de documentos.

- En [19] se examina el uso de algoritmos genéticos (GA) y programación genética (GP) para aprender algoritmos aplicados en SRI. La ponderación de la estructura del documento es una técnica mediante la cual las diferentes partes de un documento (título, resumen, etc.) contribuyen de manera desigual al peso general del documento durante la clasificación.
- En [20], este artículo describe los enfoques más destacados para aplicar IA en un SRI. El buscador de formular una consulta tratando de describir su necesidad de información. La consulta se compara con vectores de documentos que se extrajeron durante una fase de indexación. Se utiliza la función de similitud del coseno para determinar los documentos más similares, así el usuario puede evaluar la relevancia con respecto a su problema.
- Clusterdoc [21]: es un sistema de recuperación y recomendación de documentos que está dirigido a usuarios con necesidades de búsqueda de información, que a través de algoritmos de agrupamiento divide el conjunto de datos en pequeños grupos con características comunes, lo cual permite minimizar el espacio de búsqueda y proporcionar información adaptada a los intereses del usuario.

4. Caso de Estudio

Se presenta una herramienta que se programó en lenguaje C++. Un fragmento del mismo se puede ver en la Figura 5. Se empleó el entorno de desarrollo integrado (IDE), que es una aplicación de software que ayuda a los programadores a desarrollar código de software de manera eficiente, Code:Block[2]. El nombre del programa es RedDocxx.cpp. Donde xx es la versión del mismo.

```
int main( ){
    printf( "Con que prefijo quiere las
    salidas: " ) ;
    fgets( pre, 10, stdin ) ;
    if( pre[strlen(pre)-1] == '\n' )
      pre[strlen(pre)-1] = '\0' ;
      strcat( pre, "_" ) ;
      lpre = strlen( pre ) ;
    printf( "Cuantas orientaciones
    tematicas tienen los documentos?" ) ;
      scanf( "%d", &K0 ) ;
    printf( "Cuantos documentos tiene el
     corpus? " ) ;
     scanf( "%d", &Z[0].N ) ;
      do {
      printf( " Cuantos casos de
       prueba (<=5)? " ) ;
       scanf( "%d", &N_pr ) ;  }
    while( N_pr > 5 ) ;
      printf( "Puede fijar la semilla
       del azar (0=no fijar): " ) ;
       scanf( "%d", &Z[0].sem_doc ) ;
       if(Z[0].sem_doc)
       srand( Z[0].sem_doc ) ;
```

Figura 5. Segmento de código fuente, mostrando el pedido de los parámetros iniciales.

Con los datos pedidos en el inicio del programa, los cuales son:

- **Prefijo de Salida**: Prefijo que se antepondrá a los nombres que tendrán los archivos resultantes del proceso. Por ejemplo el prefijo 300K1, podría referirse a un lote de 300.000 documentos, realizado con norma 1. Elegir prefijos con semántica, facilita su manipuleo posterior, pero no es una exigencia. El cálculo no está influenciado por el prefijo. Es un recurso de orden práctico para que resultados de distintas simulaciones no se pisen en la memoria.
- **Cantidad de orientaciones temáticas**: Este valor representa la cantidad de dimensiones del vector. A este vector se lo denomina también perfil del documento.
- **Cantidad de documentos del corpus**. Si bien la experimentación esta planteada para un SRI, se podría usar para cualquier otro universo de objetos.
- **Cuantos casos de prueba**. Son las consultas que se crearan para validar la experimentación. Hubiera sido correcto en un sistema en producción pedir esto una vez armado el modelo, pero durante su gestión permitía ver resultados aun en ejecuciones abortadas por algún error.
- **Puede fijar la semilla del azar**. Al ser una simulación, se recurre al pseudoazar en diversos momentos para la generación de los perfiles, las consultas e inicializar el algoritmo Kmeans. Se utiliza la función rand() del lenguaje C++, esta arranca con semilla propia, pero puede ser cambiada. Esto permite repetir una experiencia con distinto azar. En la Figura 6 se observa como el sistema interactua en el pedido de los datos de entrada.

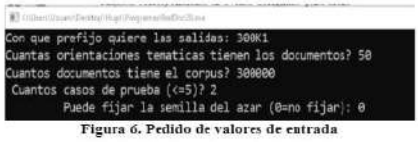


Figura 6. Pedido de valores de entrada

Luego del ingreso de estos valores iniciales, el programa crea los objetos (documentos y consultas). Después de cada (re-)distribución se actualiza el centroide tomando el centro

[2] https://www.codeblocks.org/

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

de gravedad (o sea la media) del grupo la cantidad de centroides a utilizar se solicita las siguientes preguntas:

- **Cuantos grupos quiere formar**?: En este ejemplo, para los 300.000 documentos se configuraron 30.000 grupos.
- **Puede fijar la semilla del azar para elegir semillas (0=no fijar)**: Idem al punto anterior.
- **Cuantos quiere listar en las recuperaciones:** Frente a cada pregunta, se examinará el contenido de los grupos cuyos centroides quedan más cercanos. Estos serán ordenados mediante esta distancia. Este valor limita cuantos serán examinados en detalle. Con esto se evita listados voluminosos.
- **Quiere controlar las iteraciones (S) o las deja librado al sistema (N)?:** Esta última pregunta, se realiza en todos los niveles. La información intermedia permite ver los tamaños de los grupos y da pie a cambiar parcial o totalmente las semillas del K-means. La experiencia ha demostrado que cambiarlas no introduce mejora sensible por lo tanto para producir las tablas de este escrito se ha renunciado a la posibilidad de intervenir. La segunda pregunta se refiere a la terminación. Tal como se explicó se puede dar por terminado cuando no se produzcan cambios o cuando estos estén por debajo de un umbral. Cuando queda liberado al sistema este reitera hasta la ausencia de cambios.

Por ejemplo, si el primer grupo, o sea el 0, contiene 3 documentos y estos ocupan los lugares 170, 340, 1720, entonces en una estructura "*direc*" se guardan los números 170, 340, 1720, (quedaron en orden creciente aunque ello no tiene ninguna implicación) a continuación irán los números de los que ocupan el grupo 1 y así sucesivamente. Para poder acceder rapidamente a uno de los grupos se necesita saber donde comienza y donde termina aunque queda claro que termina donde empieza el grupo siguiente. En una estructura ***QA*** (cantidades acumuladas) se guarda la posición del primer elemento de cada grupo. Y para uniformar la lógica de consulta se agrega con un grupo adicional ficticio, el fin del último grupo. Si hay K grupos hay K+1 inicios. Siempre QA[0] = 0 y QA[K] = *N* (K es la cantidad de grupos del K-means y *N* la cantidad de documentos). El programa permite establecer, varios niveles de recuperación de documentos. En el nivel siguiente *N* será la cantidad de centrodies del primer agrupamiento y K la cantidad de grupos en el segundo nivel. De cada elemento se guarda su ubicación en un vector de tamaño *N*, al cual se denomina "*está_en[]*", con lo cual se da la siguiente redundancia:

Para todo x almacenado en direc entre las posiciones QA[g] y QA[g+1] esta_en[x] = g o poniendo más variables:

Para todo x, y y g se satisface que: QA[g] <= y < QA[g+1] & direc[y] = x => esta_en[x] = q **(8)**

El sistema genera una serie de archivos, que como se comentó, llevan todos el mismo prefijo elegido por el usuario.Para este caso de estudio ***300K1***. En la Figura 7 se muesta el archivo "***300K1_resumen***".

Figura 7. Vista del archivo 300K1_resumen.

El primer dato que aparece es Norma1, recordar que una norma es una diferencia de distancia.

El archivo "***300K1_Ubicacion de cada documento en la respuesta a la consulta #1***", asigna el ordinal [1:300000] que debiera ocupar cada documento basandose en su distancia a la consulta #1. Figura 8.

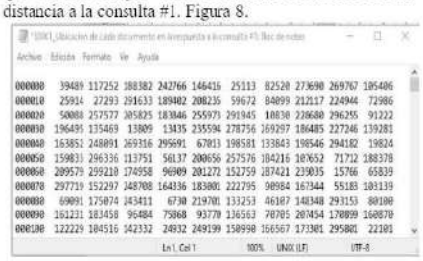


Figura 8. Vista del archivo *300K1_Ubicacion de cada documento en la respuesta a la consulta #1.*

El archivo "***300K1_fronteras de los grupos del primer nivel***", permite controlar que todo es coherente. Como hay 30000 grupos ocupa un solo renglón. Si el valor es de 5 dígitos que van desde 00000 hasta 29990. Este valor designa la ubicación en el archivo del primer elemento, por ejemplo 1380 el que está a su derecha ocupa el lugar 1381 el siguiente 1382 y así hasta el último que ocupa el lugar 1389. En esta tabla por diferencia de dos valores consecutivo se sabe el tamaño de un grupo. Figura 9.

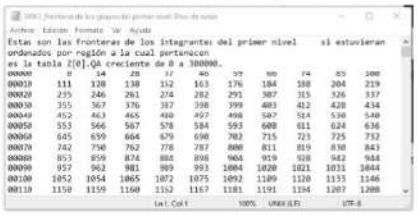


Figura 9. Vista del archivo 300K1_fronteras de los grupos del primer nivel.

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

Los "QA" se volcaron en archivos como "***300K1_Ubicacion respetando grupos del primer nivel***", "***… del segundo nivel***", etc. Figura 10.

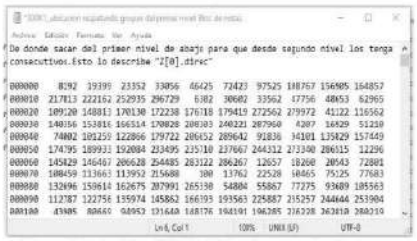


Figura 10. Archivo 300K1_Ubicacion respetando grupos del primer nivel.

Esta salida va en combinación con la figura anterior. Permite encontrar los documentos de cada grupo. La anterior te dijo que el grupo 0 tiene 14 elementos. Se puede observar que del lugar 0 al 14 hay una secuencia creciente de 14 valores. Que sea creciente no tiene ninguna importancia, es fruto del algoritmo de llenado. Detrás empieza otra secuencia con los integrantes del grupo 1.

Los "***esta_en***" se volcaron como "***300K1_grupo del primer nivel al cual pertenece cada documento***", etc. Como hay 30000 grupos en el primer nivel y alli se repartieron los 300000 documentos, en "Z[0].esta_en" se describe a que grupo pertenece cada documento.

Esta salida no depende de la consulta, servirá para verificar el contenido de los grupos. Tal vez haya tablas en exceso, fruto de en el momento de armarlas aún no se sabía exactamente cuáles eran imprescindibles.

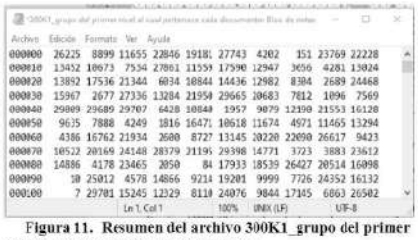


Figura 11. Resumen del archivo 300K1_grupo del primer nivel al cual pertenece cada documento.

El archivo "***300K1_Respuesta usando centroides de primer nivel a la consulta #1***", lista los documentos rescatados identificados por su ubicación en la respuesta ideal.

Figura 12. Resumen del archivo 300K1_Respuesta usando centroides de primer nivel a la consulta #1.

Por último, el archivo "***300K1_Documentos ordenados tal como van en la respuesta ideal a la consulta #1***", para que el usuario no se pierda buscando valores en tablas, donde los subindices empiezan en 0, se lista 10 documentos por fila. Esta característica la tienen todas las tablas en su formato de impresión. Si se busca el 27° documento se sabes que está en la fila encabezada por 20. Si la tabla fuera más grande habría empezdo con 000, 010, 020.

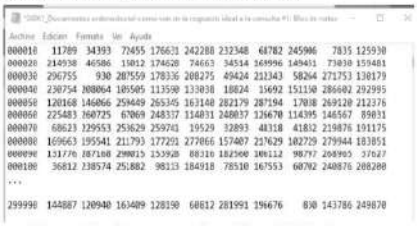


Figura 13. Resumen del archivo 300K1_ Documentos ordenados tal como van en la respuesta ideal a la consulta #1.

La idea de este primer programa, es crear una estructura se facilite a los usuarios correr diferentes consultas. Las consultas se enfrentan con todos los documentos y de acuerdo a su distancia se los ordena. En un archivo se dan los números de los documentos empezando por los más cercanos. A la inversa, por cada documento se genera la posición que ocupa en el lsitado anterior, empezando por esto y todo lo que sigue se hace para cada consulta.

Es de interés saber que pasa si se quisiera decidir la respuesta usando el primer nivel de agrupamiento. Para eso de calcula la distancia de la consulta a los K centroides y con ello se ordenan los K centroides. luego se examina el contenido de los sucesivos centroides identificando los documentos del grupo por la posición que ocupan en la respuesta ideal. Si el centroide más promisorio es por ejemplo el 7 con ayuda de QA[7] y QA[8] se delimitan los direc de los documentos que integran el grupo y consultando la tabla de posiciones de los documentos en la consulta se conocen sus números.

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

5. Resultados y discusiones

Para este trabajo se realizaron varias corridas con 50, 100, 200 y 300 mil documentos. Cada uno de estos ensayos se realizaron con las 3 normas (métrica de distancia) propuestas. Dada la cantidad de archivos resúmenes que arroja RedDocxx.cpp, fue necesario realizar otro programa para que saque, automaticamente, un resumen con la cantidad de documentos que vamos a utilizar para realizar realizar un análisis. Este programa, también se entrega, como parte del trabajo presentado, lee los siguientes archivos para realizar el resumen:

- **_resumen**
- **_ubicacion respetando grupos del primer nivel:**
- **_fronteras de los grupos del primer nivel**
- **_Respuesta usando centroides de primer nivel a la consulta #X:** Donde X es el valor ingresado como parámetro.

El archivo resultante de nombre "***300k1_k2_Extracto comprimido de primer nivel de la consulta # X***", se puede ver en la Figura 14. El valor **X,** es parte de uno de los parámetro solicitado.

Figura 14. Resumen del archivo 300K1_Extracto comprimido de primer nivel de la consulta #1.

El programa solicita entonces, una serie de entradas, que se explican a continuación:

- **prefijo de los documentos?** Como se comentó, esto fundamentalmente sirve para poder procesar diferentes lotes y no confundirlos.
- **número de la consulta?.** El sistema procesa hasta un máximo de 5 consultas. Se puede elegir cual de ellas se pretende analizar. Se recuerda que el límite surge por el hecho de procesar las mismas conforme avanzan los niveles. Si se procesara una consulta por completo, antes de procesar otra, su número sería ilimitado.
- **Hasta que ordinal de la respuesta quiere que aparezca (por ej. 100)?:** Como cada grupo entrega en promedio diez documentos, cada uno de los cuales ocupa un lugar en la respuesta ideal. Se considera que respuestas muy alejadas no ofrecen interés. Para poder cuantificar cuantas respuestas útiles se obtienen tal como ilustra la tabla 2 es necesario poner un límite. Sin un límite las figuras de barras serían imposibles de construir. 100 es un número generoso, significa que devolviendo 10 sugerencias de lectura por pantalla se podría alimentar 10 pantallas.
- **Cada cuantos milésimos quiere ver una respuesta y hasta cuanto?:** Para poder comparar sin la influencia del tamaño del corpus en los resultados, es necesario unificar la métrica. Si se fijara un valor absoluto, por ejemplo midiendo distancias a 1000 documentos, los experimentos con un corpus de 1000 documentos darían un rendimiento del 100%. Es necesario admitir que en un corpus más numeroso es inevitable mirar, o sea, medir distancias a más documentos para obtener los más cercanos.
  Se puede tomar como comparación, para entender esto la búsqueda en una tabla unidimensional ordenada, a mayor tabla el número de comparaciones aumenta, aunque sea logarítmicamente. Para este caso se ha admitido un aumento lineal si bien se espera que con el agregado de más niveles de agrupamiento llegue a ser también logarítmico.

En la siguiente Tabla se muestra el resultado de la ejecución del programa bajo una misma norma y con distintas cantidads de documentos.

Tabla 1. Cantidad de documentos recuperados bajo norma 1, con distintas cantidades de documentos. Elaboración propia.

En 0/000	Cantidad de documentos recuperados bajo norma 1 para corpus de tamaño:			
	50.000	100.000	200.000	300.000
2	8	6	20	27
4	13	14	27	37
6	17	19	31	40
8	21	25	33	44
10	24	29	35	47
12	26	33	37	47
14	30	36	39	48
16	31	38	42	51
18	34	41	43	52
20	34	41	45	54

Esta tabla permite apreciar el crecimiento conforme se miden más distancias y comparando lo recuperado entre dos líneas consecutivas se observa que conforme se avanza los agregados suelen darse preferentemente entre los ordinales más elevadas.

En un sistema en producción nunca se sabrá el ordinal de cada elemento obtenido y simplemente se lo insertará en una cola de prioridad. Con sucesivos experimentos queremos determinar una cota expresada porcentualmente en la cantidad de elmentos a examinar. Esto exigirá correr una gran cantidad de simulaciones, tarea aun no relizada, cambiando la consulta, lo que cuesta menos y cambiando

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

tanto el agrupamiento como los documentos simulados lo que implica nuevos K-means.

Entre la tabla 2, se muestra la comparativa entre las distintas normas para cada corpus.

Tabla 2. Cantidad de documentos recuperados bajo norma 1,2 y 4, en pruebas con 50. 100, 200 y 300 mil documentos.

Expresa de en 0/000	Cantidad de documentos recuperados											
	50.000			100.000			200.000			300.000		
	1	2	4	1	2	4	1	2	4	1	2	4
2	8	11	4	6	16	8	16	20	7	27	20	16
4	13	19	7	14	19	11	27	27	9	37	24	21
6	17	24	8	19	22	15	31	31	13	40	31	27
8	21	27	10	25	27	17	33	33	16	44	38	29
10	24	29	12	29	30	19	35	35	19	47	43	31
12	26	30	13	33	34	19	37	37	23	47	47	33
14	30	34	19	36	37	21	39	39	27	48	51	36
16	31	34	23	38	39	25	42	42	28	51	53	37
18	34	36	23	41	39	26	43	43	29	52	54	38
20	34	37	25	41	40	29	45	45	30	54	57	42

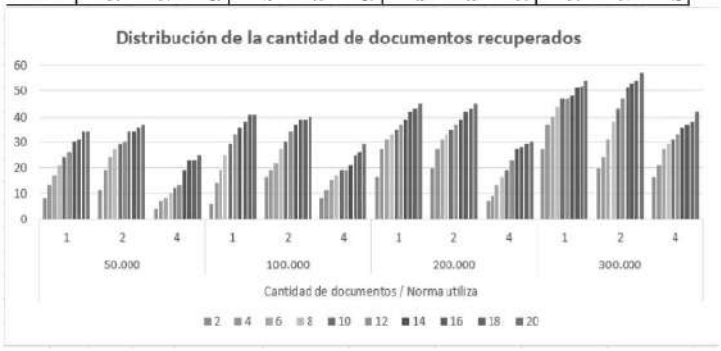


Figura 15. Gráfico de la cantidad de documentos recuperados bajo norma 1,2 y 4, en pruebas con 50, 100, 200 y 300 mil documentos. Elaboración propia.

Observando la tabla 2, se comprueba que el primer intervalo de 2 milésimos es el que aporta la mayor cantidad de elementos, lo cual confirma que ordenarlos por la distancia a los centroides es un buen criterio para examinar los contenidos. Esto queda confirmado también por la curvatura negativa cuando se expresa de modo gráfico (Figura 15). De las normas ensayadas la de orden 2, o sea la euclidiana es la que obtuvo los mejores resultados en contra de una intuición que a mayor exponente se castigaría los que presentaran puntualmente grandes diferencias en algunas de las dimensiones. A igual que la tabla anterior se ve que un corpus más numeroso mejora la calidad de la recuperación.

Se puede concluir, diciendo que no lograr el 100% de los elementos deseables no afecta a la utilidad del sistema, dada la imprecisión de la noción de similitud y la hipótesis de que contar la aparición de ciertas palabras garantiza la calidad del contenido o el acierto en la sugerencia de su lectura. Por otra parte, haber reducido el problema al agrupamiento tentativamente de a 10 obliga a un 10% de mediciones de distancia. Esta cantidad bajará drásticamente si se agrega un nivel o varios por encima repitiendo el esquema de K-means sobre los centrodies del primer nivel y así siguiendo hacia arriba. El esquema de como organizar la exploración se vuelve más complejo y será un tema a abordar en le futuro.

Universidad Tecnológica Nacional
3 y 4 de noviembre de 2022

AJEA - Actas de Jornadas y Eventos Académicos de UTN
DOI: https://doi.org/10.33414/ajea.1146.2022

6. Trabajos futuros

Es predecible que los Kmeans crezcan en froma cuadrática con el tamaño de la población. Problema similar se tuvo con los SVD que crecen en froma cúbica con la cantidad de dimensiones de los vectores [3-4]. Con aquellos se redujo drásticamente el tiempo recurriendo al paralelismo de los procesadores gráficos. Hay que tener presente que a igual que con SVD esto se calcula una vez antes de liberar el sistema y se recalcula cuando la cantidad de documentos recuperados lo justifica. Como estos recálculos se puden hacer sin afectar a la operación del sistema no tiene demasiado importancia el tiempo.

Se analizará la posibilidad de visualizar gráficamente los grupos y los centroides.

Por otro lado, se agregarán más opciones como, por ejemplo, elegir distintas métricas de distancias, o distintos algoritmos de agrupamiento (Dbscan, Modelo Mezcla Gaussiana (GMM), Equilibrio Iterativo de Reducción y Agrupación mediante Jerarquías o Balance Iterative Reducing and Clustering using Hierarchies (BIRCH por sus siglas en inglés), etc.

Finalmente, se considera que también es posible paralelizar el proceso para realizar las pruebas en lotes de mayores a un millón de documentos y con orientaciones temáticas que superen los 100 elementos.

7. Referencias

[1] Ryckeboer H., Sositto O., Bossero, J., Barone M. (2018). Recuperación de la Infomación", XX Workshop de Investigadores en Ciencias de la Computación, UNN, Corrientes, pp255-260, ISBN 978-987-3619-27-4.

[2] Sositto, O., Procopio, G. Bossero, J. (2018). "Método para la Construcción de un Corpus Periodístico mediante Expresiones Regulares". CONAIISI 2018, pp. 491-500. Universidad CAECE, Mar del Plata, 2018

[3] Sositto, O., Casuscelli, M., Bossero, J., Matteo, L., Ryckeboer, H. (2018). "Aceleración en la Recuperación de Información utilizando Algoritmos de Minería de Datos de R". CACIC 2018, pp.491-500.Universidad Nacional del Centro, Tandil.

[4] Sositto O., Ledesma V. y Procopio G. (2019). "Aplicación de la Descomposición de Valores Singulares a un Sistema de Recuperación de Información", REDDI, Revista Digital del Departamento de Ingeniería (UNLaM), volumen 2(4),ISSN: 2525-1333.

[5] Sositto O., Ledesma V., Procopio G. y Ryckeboer H. (2020). "Hacia la Optimización de un Sistema de Recuperación de Información", XXII Workshop de Investigadores en Ciencias

[18] Chris Piech.(2012). K Means. https://stanford.edu/~cpiech /cs221/handouts/kmeans.html

[19] Trotman, A. (2004). An artificial intelligence approach to information retrieval (abstract only). Proceedings of the 27th Annual International Conference on Research and Development in Information Retrieval - SIGIR '04. https://doi.org/10.1145/1008992.1009150

de la Computación (WICC 2020), El Calafate, Santa Cruz, pp. 738-742, mayo 2020. ISBN 978-987-3714-82-5.

[6] Sositto O., Ledesma V., Procopio G., Ryckeboer H., Saizar V. y Vainberg A. (2020). "Comparación de un Algoritmo de Bidiagonalización para su Utilización en la Recuperación de Información", XXVI Congreso Argentino de Ciencias de la Computación (CACIC 2020). Universidad Nacional de La Matanza, pp. 72-81. Modalidad virtual, octubre de 2020. ISBN 978-987-4417-90-9.

[7] Sositto O., Ledesma V., Procopio G., Saizar V. (2020). "Implementación de un Algoritmo de Bidiagonalización en un Entorno Híbrido para su Aplicación en la Recuperación de Información", 8vo. Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2020). U TN, San Francisco, Córdoba, noviembre de 2020.

[8] Hernández Orallo, J. y otros. "Introducción a la minería de datos". Editorial: Pearson. Edición: 1. Año 2004

[9] Giubergia, A y otros. Modelado y Diseño de Sistemas Complejos mediante Técnicas de Simulación. WICC 2014 XVI Workshop de Investigadores en Ciencias de la Computación. Disponible en: https://core.ac.uk/download/pdf/296374584.pdf

[10] "Introduction to Modern Information Retrieval". Gerard Salton, Michael J. Michael J. McGill. Ed. McGraw-Hill, Inc. New York, NY, USA. ISBN: 0070544840. 1986

[11] Tolosa Gabriel H. y Bordignon Fernando R.A. "Introducción a la Recuperación de Información - Conceptos, modelos y algoritmos básicos". Universidad Nacional de Lujan. Creative Commons Atribución-No Comercial-Compartir Obras Derivadas Igual 2.5 Arg. License. 2007. Disponible en. http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf

[12] Modelo vectorial de recuperación de información. (2020, febrero 26). EcuRed, Disponible en: https://www.ecured.cu/index.php?title=Modelo_vectorial_de_recuperaci%C3%B3n_de_informaci%C3%B3n&oldid=3644075.

[13] Larson, Ron. (2013). Fundamentos de álgebra lineal, séptima edición ISBN: 978-607-519-803-3

[14] Dos Santos, E. & otros. (2015). Procesamiento de búsquedas por similitud. Tecnologías de paralelización e indexación. ICT-UNPA-115-2015 ISSN: 1852-4516

[15] Guccione, Jorge y Guccione, J. (2018) Espacios Métricos. UBA http://cms.dm.uba.ar/academico /materias/1ercuat2014/calculo_avanzado/ EspaciosMetricos.pdf

[16] Reyes Nora Susana. (2015). Bases de Datos Métricas. Universidad Nacional de San Luis. Disponible en: https://users.dcc.uchile.cl/~gnavarro/algoritmos/tesisNora.pdf

[17] "Inteligencia artificial avanzada". Raúl Benitez, Gerard Escudero, Samir Kanaan. Universitat Oberta de Catalunya. 2013. Disponible en: http://archivos.inteligencia-artificial.net/archivos/RaulBenitezyotros.%20Inteligencia%20Artificial%20Aumentada.pdf

[20]Mandl, Thomas. (2008). Artificial Intelligence for Information Retrieval. 10.4018/9781599048499.ch023.

[21] Giugni O, Marylin & LEON G, Luis. Clusterdoc un sistema de recuperación y recomendación de documentos basado en algoritmos de agrupamiento. uct [online]. 2011, vol.15, n.60, pp.121-129. ISSN 1316-4821.

ANEXO I

CERTIFICADOS DIVULGACIÓN

ReDDI

San Justo, 3 de enero de 2023

Certificamos que el artículo

"Determinación del umbral inferior de coincidencia aplicando medidas de edición a términos jurídicos",

de Lorena Romina Matteo, Viviana Ledesma y Osvaldo Spositto, ha sido publicado en el Volumen: 7 - Número 2 (Diciembre-2022) de la revista digital ReDDI ISSN: 2525-1333.

Dra. Bettina Donadello
Gestión Editorial
ReDDI E-Journal

Mg. Jorge Eterovic
Dirección Ejecutiva
ReDDI E-Journal

Universidad Nacional de La Matanza

DIIT

ISSN: 2525-1333

CACIC 2021

UNIVERSIDAD NACIONAL DE SALTA

RedUNCI

XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Por medio del presente se CERTIFICA que:

Edgardo J. Moreno

Ha participado en calidad de AUTOR del trabajo "Construcción de un Corpus Jurídico utilizando Expresiones Regulares" en el XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN llevado a cabo de manera virtual por la Facultad de Ciencias Exactas de la UNSa del 04 al 08 de octubre de 2021.

Salta, Argentina

Firmado digitalmente por: GIL Gustavo Daniel
Motivo: Vice Decano - Facultad de Ciencias Exactas
Fecha y hora: 20.10.2021 12:06:35

Lic. Patricia Pesado
Coordinadora
Red UNCI

Ing. Daniel Hoyos
Decano
Facultad de Ciencias Exactas
UNSa

CACIC 2021

UNIVERSIDAD NACIONAL DE SALTA

RedUNCI

XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Por medio del presente se CERTIFICA que:

Hugo Emilio Ryckeboer

Ha participado en calidad de AUTOR del trabajo "Construcción de un Corpus Jurídico utilizando Expresiones Regulares" en el XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN llevado a cabo de manera virtual por la Facultad de Ciencias Exactas de la UNSa del 04 al 08 de octubre de 2021.

Salta, Argentina

Firmado digitalmente por: GIL Gustavo Daniel. Motivo: Vice Decano - Facultad de Ciencias Exactas. Fecha y hora: 20.10.2021 12:06:52

Lic. Patricia Pesado
Coordinadora
Red UNCI

Ing. Daniel Hoyos
Decano
Facultad de Ciencias Exactas
UNSa

CACIC 2021

UNIVERSIDAD NACIONAL DE SALTA

RedUNCI

XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Por medio del presente se CERTIFICA que:

julio Bossero

Ha participado en calidad de AUTOR del trabajo "Construcción de un Corpus Jurídico utilizando Expresiones Regulares" en el XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN llevado a cabo de manera virtual por la Facultad de Ciencias Exactas de la UNSa del 04 al 08 de octubre de 2021.

Salta, Argentina

Firmado digitalmente por: GIL Gustavo Daniel. Motivo: Vice Decano - Facultad de Ciencias Exactas. Fecha y hora: 20.10.2021 12:07:16

Lic. Patricia Pesado
Coordinadora
Red UNCI

Ing. Daniel Hoyos
Decano
Facultad de Ciencias Exactas
UNSa

CACIC 2021

UNIVERSIDAD NACIONAL DE SALTA

RedUNCI

XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Por medio del presente se CERTIFICA que:

Lorena Matteo

Ha participado en calidad de AUTOR del trabajo "Construcción de un Corpus Jurídico utilizando Expresiones Regulares" en el XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN llevado a cabo de manera virtual por la Facultad de Ciencias Exactas de la UNSa del 04 al 08 de octubre de 2021.

Salta, Argentina

Firmado digitalmente por: GIL Gustavo Daniel
Motivo: Vice Decano - Facultad de Ciencias Exactas
Fecha y hora: 20.10.2021 12:07:35

Lic. Patricia Pesado
Coordinadora
Red UNCI

Ing. Daniel Hoyos
Decano
Facultad de Ciencias Exactas
UNSa

CACIC 2021

UNIVERSIDAD NACIONAL DE SALTA

RedUNCI

XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Por medio del presente se CERTIFICA que:

Osvaldo Mario Spositto

Ha participado en calidad de AUTOR del trabajo "Construcción de un Corpus Jurídico utilizando Expresiones Regulares" en el XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN llevado a cabo de manera virtual por la Facultad de Ciencias Exactas de la UNSa del 04 al 08 de octubre de 2021.

Salta, Argentina

Firmado digitalmente por: GIL Gustavo Daniel
Motivo: Vice Decano - Facultad de Ciencias Exactas
Fecha y hora: 20.10.2021 12:07:52

Lic. Patricia Pesado
Coordinadora
Red UNCI

Ing. Daniel Hoyos
Decano
Facultad de Ciencias Exactas
UNSa

CACIC 2021

UNIVERSIDAD NACIONAL DE SALTA

RedUNCI

XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Por medio del presente se CERTIFICA que:

Viviana Ledesma

Ha participado en calidad de AUTOR del trabajo "Construcción de un Corpus Jurídico utilizando Expresiones Regulares" en el XXVII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN llevado a cabo de manera virtual por la Facultad de Ciencias Exactas de la UNSa del 04 al 08 de octubre de 2021.

Salta, Argentina

Firmado digitalmente por: GIL Gustavo Daniel
Motivo: Vice Decano - Facultad de Ciencias Exactas
Fecha y hora: 20.10.2021 12:08:39

Lic. Patricia Pesado
Coordinadora
Red UNCI

Ing. Daniel Hoyos
Decano
Facultad de Ciencias Exactas
UNSa

CERTIFICADO

Julio Cesar Bossero, Osvaldo Mario Spositto, Viviana Ledesma, Hugo Ryckeboer, Laura Conti, Sergio Garcia, Edgardo Moreno, Lorena Matteo, Victoria Saizar, Patricio Macias, Fabio Quintana, Gustavo Perez Villar, Cecilia Gargano, Gaston Procopio,

han presentado el trabajo

en el **9º Congreso Nacional de Ingeniería Informática / Sistemas de Información (CoNaIISI 2021)** organizado por la Red de Carreras de Ingeniería Informática / Sistemas de Información (RIISIC) perteneciente al CONFEDI, realizado de forma Virtual por la Universidad Tecnológica Nacional Facultad Regional Mendoza, los días 04 y 05 de noviembre de 2021; se otorga el presente certificado en calidad de autor.

Ing. Nelson Roberto Sotomayor
Coordinador RIISIC 2021

Mg. Ing. Marcela Fernandez
Coordinadora CoNaIISI 2021

Esp. Ing. José Balacco
Decano UTN - FRM

UTN Departamento de Ingeniería en Sistemas de Información

RIISIC

9° CoNaIISI 2021

confedi

WICC 2022 MENDOZA — UNIVERSIDAD CHAMPAGNAT — RedUNCI

XXIV WORKSHOP DE INVESTIGADORES EN CIENCIAS DE LA COMPUTACIÓN

Se certifica que

Julio C. Bossero

ha participado en calidad de autor del artículo

Adecuación de un Sistema de Recuperación de Información para su Utilización en un Contexto Jurídico

aceptado en el XXIV WORKSHOP DE INVESTIGADORES EN CIENCIAS DE LA COMPUTACIÓN – WICC 2022 organizado por la Universidad Champagnat.

Abril 2022 - Mendoza, Argentina.

Lic. Patricia Pesado
Coordinadora
RedUNCI

Lic. Alejandro Giuffrida
RECTOR
UNIVERSIDAD CHAMPAGNAT

WICC 2022 MENDOZA — UNIVERSIDAD CHAMPAGNAT — RedUNCI

XXIV WORKSHOP DE INVESTIGADORES EN CIENCIAS DE LA COMPUTACIÓN

Se certifica que

Viviana Ledesma

ha participado en calidad de autor del artículo

Adecuación de un Sistema de Recuperación de Información para su Utilización en un Contexto Jurídico

aceptado en el XXIV WORKSHOP DE INVESTIGADORES EN CIENCIAS DE LA COMPUTACIÓN – WICC 2022 organizado por la Universidad Champagnat.

Abril 2022 - Mendoza, Argentina.

Lic. Patricia Pesado
Coordinadora
RedUNCI

Lic. Alejandro Giuffrida
RECTOR
UNIVERSIDAD CHAMPAGNAT

Certificado

UTN FRCU

10mo CONGRESO NACIONAL CoNaIISI 2022

Osvaldo Sposito, Julio Bossero, Edgardo Moreno, Viviana Ledesma, Gastón Procopio, Lorena Matteo, Cecilia Gargano, Victoria Saizar, Patricio Macias, Juan Ojeda, Fabio Quintana, Laura Conti, Sergio García, Gustavo Pérez Villar

han participado como autores del trabajo titulado "Resultados preliminares de una técnica de localización de documentos en espacios métricos utilizando K-means" y que el mismo ha sido aceptado para ser presentado en el 10° Congreso Nacional de Ingeniería Informática / Sistemas de Información -CONAIISI 2022- organizado por la Red de Carreras de Ingeniería Informática / Sistemas de Información -RIISIC- perteneciente al CONFEDI, realizado de forma híbrida por la Facultad Regional Concepción del Uruguay de la Universidad Tecnológica Nacional, los días 3 y 4 de noviembre de 2022.

Concepción del Uruguay, Entre Ríos, 4 de noviembre de 2022.-

Lic. Augusto Nasrallah
COORDINADOR RIISIC 2022

Mg. Patricia Cristaldo
COORDINADORA CONAIISI 2022

Esp. Ing. Martín E. Herlax
DECANO UTN FRCU

RIISIC confedi

Red UNCI

UNLaR DACEFyN Universidad Nacional de La Rioja

cacic2022

Se certifica que Osvaldo Sposito, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del **XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022)**, realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-

Lic. Patricia Pesado
Coordinador titular
Red UNCI

Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Luis Busnelli, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Viviana Ledesma, ha participado en calidad de Autora del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales

Red UNCI

UNLaR DACEFyN | Universidad Nacional de La Rioja

cacic2022

Se certifica que Cecilia Gargano, ha participado en calidad de Autora del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del **XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022)**, realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-

Lic. Patricia Pesado
Coordinador titular
Red UNCI

Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales

Red UNCI

UNLaR DACEFyN | Universidad Nacional de La Rioja

cacic2022

Se certifica que Julio Bossero, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del **XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022)**, realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-

Lic. Patricia Pesado
Coordinador titular
Red UNCI

Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Gerardo Frega, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Victoria Saizar, ha participado en calidad de Autora del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Fabio Quintana, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Laura Conti, ha participado en calidad de Autora del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Sergio García, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales


Red UNCI
UNLaR DACEFyN | Universidad Nacional de La Rioja
cacic2022
Se certifica que Carlos Colombain, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022), realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-
Lic. Patricia Pesado
Coordinador titular
Red UNCI
Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales

Red UNCI

UNLaR DACEFyN Universidad Nacional de La Rioja

cacic2022

Se certifica que Gustavo Pérez Villar, ha participado en calidad de Autor del artículo: Desarrollo de Interfaces de Programación de Aplicaciones aplicadas en Experticia, un Sistema Experto Jurídico; aceptado en el marco del **XXVIII Congreso Argentino de Ciencias de la Computación - (CACIC 2022)**, realizado en la ciudad de La Rioja entre los días 03 al 06 de octubre del 2022.-

Lic. Patricia Pesado
Coordinador titular
Red UNCI

Lic. Miguel A. Molina
Decano del Departamento de
Ciencias Exactas Físicas y Naturales

51 JAIIO
JORNADAS ARGENTINAS DE INFORMÁTICA

17 AL 27 DE OCTUBRE 2022

SADIO
Sociedad Argentina de Informática

extiende el presente diploma a

VIVIANA ALEJANDRA LEDESMA

por haber participado como **Panelista** de

SID 2022 - Simposio Argentino de Informática y Derecho

durante las 51° Jornadas Argentinas de Informática, realizadas del 17 al 27 de Octubre de 2022.

Claudia Pons
Coordinadora General de las 51 JAIIO

Marcelo De Vincenzi
Coordinador General de las 51 JAIIO

Carlos Neil
Coordinador General de las 51 JAIIO

Sandra D´Agostino
Presidenta de SADIO

51jaiio.sadio.org.ar

51 JAIIO
JORNADAS ARGENTINAS DE INFORMÁTICA

17 AL 27 DE OCTUBRE 2022

SADIO
Sociedad Argentina de Informática

extiende el presente diploma a

VIVIANA ALEJANDRA LEDESMA

por haber participado como **Panelista** de

SID 2022 - Simposio Argentino de Informática y Derecho

durante las 51° Jornadas Argentinas de Informática, realizadas del 17 al 27 de Octubre de 2022.

Claudia Pons
Coordinadora General de las 51 JAIIO

Marcelo De Vincenzi
Coordinador General de las 51 JAIIO

Carlos Neil
Coordinador General de las 51 JAIIO

Sandra D´Agostino
Presidenta de SADIO

51jaiio.sadio.org.ar

Certificado

FUNDEJUS

Fundación de Estudios para la Justicia

Pers. Jur. Nº 10.864 - Dir. Pers. Jur. Pcia. Bs. As.
Organización No Gubernamental reconocida por la ONU

CERTIFICO: que Laura Conti, ha participado en carácter de disertante en la Jornada de actualización *" La gestión judicial en los tiempos actuales: nuevos desafíos "*, llevada a cabo el día miércoles 2 del corriente mes y año, organizada por FUNDEJUS (Fundación de Estudios para la Justicia) y la Asociación de Magistrados y Funcionarios del Departamento Judicial de Morón en el marco del convenio de cooperación firmado por ambas instituciones. Auspiciada por el Colegio de Abogados del Departamento Judicial de Morón.-

Expido el presente, a pedido de la interesada, para ser presentado por ante quien corresponda, a los 8 días del mes de noviembre del año 2022.-

Carlos Fabián Blanco
VICE-PRESIDENTE DE FUNDEJUS

Lavalle 1580 4º E (C1048AAL) Ciudad Autónoma de Bs. As.
Tel./Fax: (54-011) 4374-6616
http://www.fundejus.org *E-mail: info@fundejus.org*
República Argentina

ANEXO I

CERTIFICADOS CAPACITACIÓN

EDUCACIÓN IT

Certificado
de Asistencia

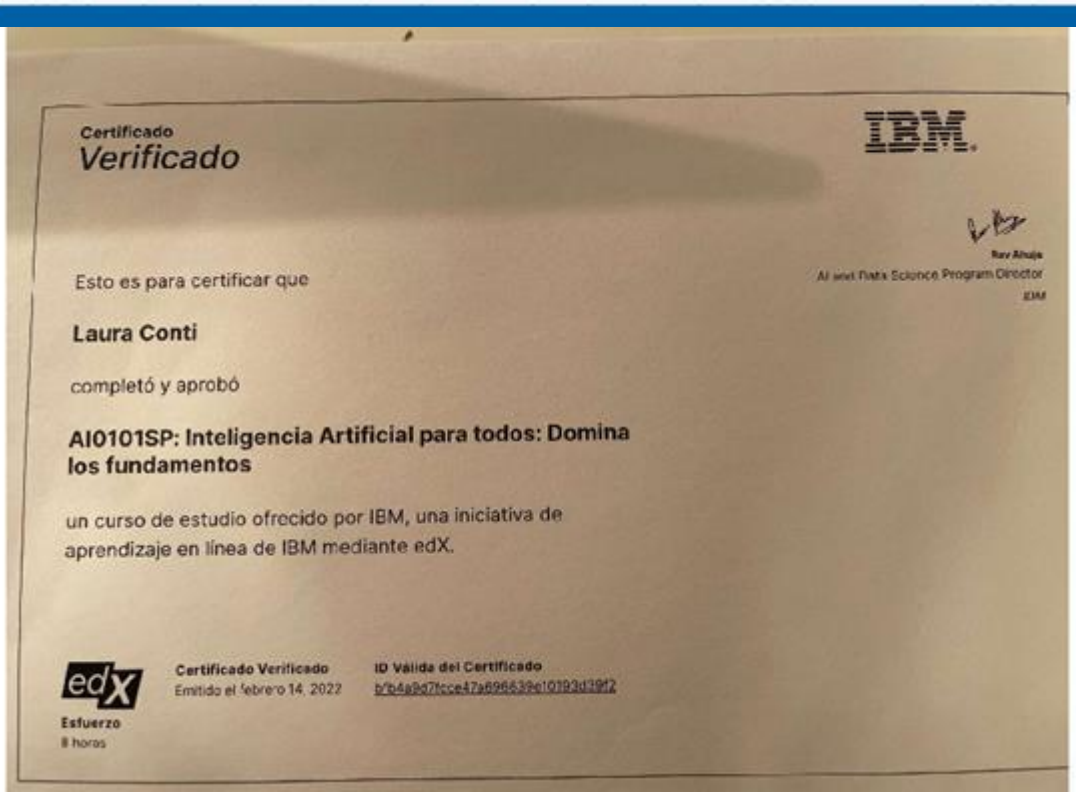
Julio Bossero

Ha asistido y completado el curso:

Git: Desarrollo Colaborativo

9/12/2021
Fecha de emisión

Lorena Somer
Coordinadora Academica



Certificado
Verificado

IBM

Esto es para certificar que

Laura Conti

completó y aprobó

AI0101SP: Inteligencia Artificial para todos: Domina los fundamentos

un curso de estudio ofrecido por IBM, una iniciativa de aprendizaje en línea de IBM mediante edX.

edX

Certificado Verificado
Emitido el febrero 14, 2022

ID Válida del Certificado

Esfuerzo
8 horas

doinGlobal

Online smart learning

CONSTANCIA DE FINALIZACIÓN DE CURSADO – CERTIFICADO EN TRÁMITE

Se deja constancia que la alumna Conti, Laura, legajo número 15386, número de identificación 24028816, actualmente ha finalizado el cursado de manera regular y aprobado el Curso Superior en Derecho: Inteligencia Artificial y Derecho, dictado y certificado por la Fundación General de la Universidad de Salamanca (Salamanca – España) & doinGlobal (Silicón Valley - USA), encontrándose a la fecha el certificado correspondiente en proceso de expedición.

Cantidad de horas cátedra curso: 120 horas.

A solicitud de la interesada y para ser presentado ante quien corresponda, se expide esta constancia a los veintisiete días del mes de diciembre del año dos mil veintidós.

Belén Redondo
Oficina de Alumnos

[1] doinGlobal
5493515 52-2850 - belen.redondo@doinglobal.com
2880 Zanker Road - San Jose, CA 95134 -Silicon Valley | USA
doinglobal.com

EDUCACIÓN IT

Certificado de Aprobación

Cecilia, Gargano

Ha completado y aprobado el curso:
Python Programming

Duración:
21hs

Lorena Sommer
Coordinadora Académica

04/11/21
Fecha

EDUCACIÓN IT

Certificado de Aprobación

Cecilia, Gargano

Ha completado y aprobado el curso:
Python para Analisis de Datos

Duración:
18hs

Lorena Sommer
Coordinadora Académica

04/02/22
Fecha

EDUCACIÓN IT

Certificado de Aprobación

Viviana, Ledesma

Ha completado y aprobado el curso:
Introducción a UX

Duración:
12hs

Lorena Sommer
Coordinadora Académica

21/10/21
Fecha

EDUCACIÓN IT

Certificado
de Asistencia

Viviana Ledesma

Ha asistido y completado el curso:

Diseño de interfaz de usuario

29/11/2021
Fecha de emisión

Lorena Somer
Coordinadora Academica

REPUBLICA ARGENTINA

Universidad Nacional de La Matanza

Dirección de Pedagogía Universitaria

Se deja constancia que **Matteo, Lorena Romina, DNI 23701391** ha asistido al intercambio pedagógico - didáctico "*Aulas híbridas y bimodalidad en la educación superior*", en esta Casa de Altos Estudios. Por tanto, se extiende el presente Certificado.

San Justo, 28 de octubre de 2021

Lic. Jorgelina Monti
Directora de Pedagogía Universitaria

biblioteca
LEOPOLDO MARECHAL

CERTIFICADO

Buenas prácticas de consumo y producción de la información científica y académica

LORENA MATTEO

Certificamos que ha participado en calidad de asistente al ciclo de capacitación organizado y dictado por la Biblioteca Leopoldo Marechal de la Universidad Nacional de La Matanza

03 de noviembre de 2022

Lic. Bib. Daniela Rodríguez
A/C Directora
Biblioteca Leopoldo Marechal
Universidad Nacional de La Matanza

educación IT

Certificado de aprobación

Edgardo Moreno

Ha asistido y completado el Curso:

Programación .NET con C# .NET

Duración
40 h

16 de mayo de 2022
Fecha de emisión

Lorena Somer
Coordinadora Académica

educación IT

Certificado de aprobación

Edgardo Moreno

Ha asistido y completado el Curso:

Programación .NET con C# .NET

Duración
40 h

16 de mayo de 2022
Fecha de emisión

Lorena Somer
Coordinadora Académica

EDUCACIÓN IT

Certificado
de Asistencia

Fabio Quintana

Ha asistido y completado el curso:

Node.js y Mongo DB

14/1/2022
Fecha de emisión

Lorena Somer
Coordinadora Academica

EDUCACIÓN IT

Certificado
de Asistencia

Fabio Quintana

Ha asistido y completado el curso:

Vue.JS

4/12/2021
Fecha de emisión

Lorena Somer
Coordinadora Academica

EDUCACIÓN IT

Certificado de Aprobación

Victoria, Saizar Godoy

Ha completado y aprobado el curso:
ReactJS Developer

Duración:
36hs

Lorena Sommer
Coordinadora Académica

07/07/22
Fecha

EDUCACIÓN IT

Certificado de Aprobación

Victoria, Saizar Godoy

Ha completado y aprobado el curso:
Web API .Net Core

Duración:
21hs

Lorena Sommer
Coordinadora Académica

06/07/22
Fecha

UNLaM - SECyT Programa PROINCE FI-013 **FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN**

Unidad Académica: **Departamento de Ingeniería e Investigaciones Tecnológicas**
Código: **C241**
Título del Proyecto: **Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado**
Director del Proyecto: Sposito, Osvaldo
Fecha de inicio: **1/1/2021**
Fecha de finalización**: 31/12/2022**

---

1. Datos del alumno
Apellido y Nombre: **Juan Ojeda**
DNI: **32.265.949**
Unidad Académica: **Departamento de Ingeniería e Investigaciones Tecnológicas**
Carrera que cursa: **Ingeniería en Informática**
Período evaluado: **1/1/2022 a 31/12/2022**

2. Dictamen de evaluación de desempeño del alumno:
Colocar una cruz donde corresponda

2.1 Satisfactorio: X
2.1 No satisfactorio:

Fundamentos del dictamen:
El alumno trabajó de modo satisfactorio, desenvolviéndose con idoneidad en la realización de sus actividades definidas para el primer año del cronograma.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)
Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto:
3.2 No continuar en el presente proyecto:

Fundamentos del dictamen:
El alumno ha mostrado responsabilidad y dedicación en su trabajo.

San Justo 31/12/2022		Sposito Osvaldo Mario
.....	.....	.....
Lugar y fecha	Firma del Director	Aclaración de firma

Unidad Académica: **Departamento de Ingeniería e Investigaciones Tecnológicas**
Código: **C241**
Título del Proyecto: **Implementación de un Sistema Web de Recuperación de la Información Orientado a Documentación Jurídica con el Proceso de Indexación Semántica Latente Paralelizado**
Director del Proyecto: Sposito, Osvaldo
Fecha de inicio: **1/1/2021**
Fecha de finalización**: 31/12/2022**

---

1. Datos del alumno
Apellido y Nombre: **Fabio Quintana**
DNI: **33.676.620**
Unidad Académica: **Departamento de Ingeniería e Investigaciones Tecnológicas**
Carrera que cursa: **Ingeniería en Informática**
Período evaluado: **1/1/2022 a 31/12/2022**

2. Dictamen de evaluación de desempeño del alumno:
Colocar una cruz donde corresponda

2.1 Satisfactorio: X
2.1 No satisfactorio:

Fundamentos del dictamen:
El alumno se desempeñó satisfactoriamente en la realización de sus actividades planificadas en el cronograma para el primer año del proyecto.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)
Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto:
3.2 No continuar en el presente proyecto:

Fundamentos del dictamen:
El alumno ha mostrado responsabilidad y dedicación en su trabajo. Es una persona respetuosa y de buen trato.

San Justo 31/12/2022		Sposito Osvaldo Mario
.....	.....	.....
Lugar y fecha	Firma del Director	Aclaración de firma