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:

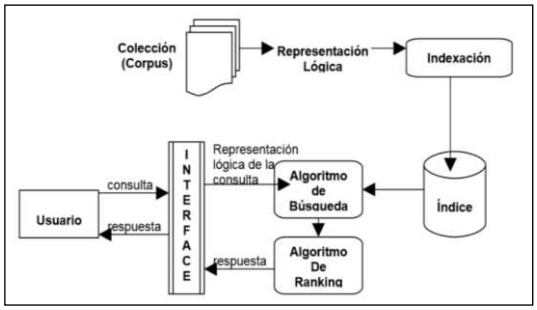

Fig. 1 – Arquitectura de un SRI. Fuente [6]

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 3º de la Ley Nº 25.188; art. 47 del Decreto 41/99 y art. 30 de la Ley Nº 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 capitulo 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].

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", "do$"; 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).

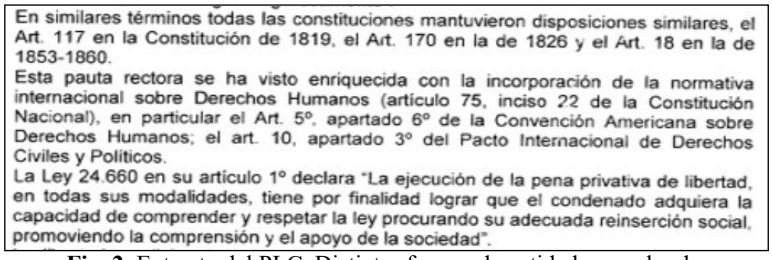
En similares términos todas las constituciones mantuvieron disposiciones similares, el Art. 117 en la Constitución de 1819, el Art. 170 en la de 1826 y el Art. 18 en la de 1853-1860.
Esta pauta rectora se ha visto enriquecida con la incorporación de la normativa internacional sobre Derechos Humanos (artículo 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 Línea: 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 Científicas 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.