Análisis exploratorio de sentencias tipo de la tutela corte constitucional colombiana

Jorge Leonardo Raba, Oscar Alexander Mendez
Universidad Nacional Bogotá, Colombia
Email: jrabag@unal.edu.co, omendeza@unal.edu.co

Figura 1: Escudo de la Universidad.

***Resumen*—Para ayudar al entendimiento de los datos se realiza un análisis exploratorio sobre las sentencias tipo tutela, donde la mayor cantidad de información se encuentra en el texto, y dado la naturaleza semi-estructurada de los documentos legales, se usarán estas secciones para analizar las palabras que componen cada una de estas secciones.**

***Index Terms*—natural language processing, judgement, data exploration, legal text**

I. Introducción

La corte constitucional colombiana es la entidad encargada de la protección de los derechos fundamentales, está se encuentra integrada por 9 magistrados, donde mensualmente se hace una selección aleatoria de las tutelas de todo el país para hacer una revisión de las sentencias.

En un año más de 600.000 mil tutelas pueden ser radicas en esta entidad, sin embargo menos de dos mil sentencias son falladas en un año[1], por lo que es de vital importancia la selección de tutelas que se serán asignadas a los magistrados, pues con esto no solo se protegen los derechos de las personas, sino que también se genera nueva jurisprudencia para que desde las primeras instancias en las cuales una tutela es interpuesta pueda ser fallada teniendo en cuenta casos anteriores.

Las sentencias falladas responden a la razón de unos mínimos necesarios de argumentación, los cuales son presentados en los antecedentes. La razón de la decisión es un argumento, por lo cual, extraer el argumento consiste en determinar cuáles son las premisas que lo constituyen, sin necesidad de copiar y pegar apartes de la decisión a partir de premisas simples que constituyen el argumento jurídico. Este argumento es utilizado por la Corte para tomar la decisión, la cual se sustenta en las normas jurídicas relevantes que amparan el derecho que ha sido tutelado, donde la norma principal es la constitución política, soportándose en tratados internacionales y fallos de sentencias previas centradas en la protección de derecho a la vida digna.

De forma general, en la formulación del problema jurídico se pueden identificar elementos como el demandante, demandado y las pretensiones, donde a partir de los hechos jurídicamente relevantes, se narra la situación que da lugar a una demanda y se presentan los elementos probatorios que serán analizados a la luz de las normas o derechos vulnerados.

Los elementos que ayudan a identificar los elementos de una tutela son: el magistrado ponente, la sala de decisión, los magistrados que aclaran voto, los magistrados que salvan voto, los hechos relevantes, el problema jurídico enunciado por la corte, las normas jurídicas relevantes para el caso y la decisión de la corte [1].

En el texto de la sentencia se pueden identificar estos elementos de identificación, además cuando la sentencia es fallada, esta es cargada en la plataforma de la corte constitucional de Colombia, con el objetivo de permitir su búsqueda a partir de algunos parámetros, donde se agregan datos de la relatoría como el tema, la fecha de la sentencia, la fecha de la relatoría, el expediente, el magistrado ponente, el demandante y el demandado.

Dado que los elemento más representativos se encuentran en el texto, el descubrimiento del conocimiento se enfoca especialmente en el análisis de los texto a través de representaciones de vectores, extracción de normas por medio de expresiones regulares, lo cual permite evidenciar algunas relaciones entre los derechos tutelados y las normas usadas para soportarlo. Además, por medio del análisis de los vectores junto con el texto se pueden descubrir algunas de las temáticas más tratadas por la corte.

II. Metodología

Para realizar la minería de datos, se usarán los pasos planteados en el proceso de KDD (Knowlegde Data Discovery) [2] y para la segunda etapa se aplicarán 8 de los 9 pasos del proceso.

1. **Aprendizaje del dominio de aplicación**: El texto de la sentencia esta compuesto principalmente de 3 secciones, primero los hechos relevantes, luego el problema jurídico enunciado por la corte, y por último la decisión de la corte. En cada tutela se debe identificar los derechos que fueron vulnerados y las normas jurídicas que soportan la protección de ese derecho.
2. **Crear un conjunto de datos objetivo**: El conjunto de datos se centrará sobre el análisis de texto de la

[1] https://www.corteconstitucional.gov.co/lacorte/estadisticas.php

sentencia. Los datos que son agregados después que la sentencia es fallada tendrán menor prioridad al momento de realizar los análisis, sin embargo estos datos son de gran importancia al momento de un análisis asociativo, entre las los derechos vulnerados y las leyes que protegen esos derechos.

3. **Limpieza y preprocesamiento de datos**: Para el preprocesamiento del texto, este será normalizado eliminado caracteres propios del español como las tildes, el texto se transformará a minúscula y no se tendrán en cuenta los signos de puntuación, números, ni palabras con menos de 3 caracteres.
4. **Reducción y proyección**: El texto pre-procesado será representado por el método de embedding llamado Distributed Memory Model of Paragraph Vectors (PV-DM) [3] , el cuál permite representar un documento manteniendo la semántica de las palabras en el contexto que estas son usadas.
5. **Empatar las metas a un método particular de minería de datos**: Para buscar algunas relaciones entre los derechos tutelados y las leyes que los protegen, se usarán reglas de asociación, algoritmos de agrupamiento y algoritmos de clasificación.
6. **Análisis exploratorio y selección de modelo e hipótesis**: La selección de un método de clustering se hará tomando en cuenta el mayor coeficiente de silueta [4] entre los algoritmos K-means, Gaussian mixure y DBScan. Dado que para la clasificación se tendrá un vector, se usaran métodos enfocados con el tratamiento de datos continuos como Support Vector Machine, regresiones lineales y modelos Gaussinanos, junto con técnica de ensamble.
7. **Minería de datos**: La búsqueda de patrones se hará usando la representación vectorial y la representación de la sentencia en términos de derechos y normas. Con la representación obtenida de cada documentos y usando el método de agrupamiento con mejores resultados se determina cual a grupo pertenece cada una se las sentencias y analizar el texto más representativo de cada uno de los grupo obtenidos. Con la representación por derechos y normas permite realizar un análisis asociativo por medio de técnicas para la generación de reglas.
8. **Interpretación de patrones minados**: Los resultados del mejor modelo de clustering se realiza por medio medio de una nube de palabras para cada uno de los grupos identificados, y la coherencia de las reglas de asociación se realiza evaluando las normas asociadas a cada a los derechos.

II-A. Aprendizaje del dominio de aplicación

Las solicitudes de tutela que llegan a la corte cubre una gran variedad participantes, dado que la vulneración de derechos puede ser generada desde una persona hasta una norma y los afectados pueden ser tanto individuos como organizaciones, por lo cual, se pueden distinguir diferentes tipos de sentencias tal como las tipo **C**, **T** y **SU**.

Las sentencias tipo C, son sentencias donde se tutelan leyes que van en contra de los derechos contemplados en la constitución. Las tipos SU son tutelas donde participan los 9 magistrados y lo que se busca es la unificación de conceptos entre las corte constitucional y el consejo de estado. Las sentencias tipo T son tutelas donde las personas buscan restaurar un derecho fundamental que ha sido vulnerado, a estas sentencias se les asigna un magistrado ponente y dos magistrados más. En todas las tutelas, los elementos estructurales que la distinguen en cuanto a su contenido son los antecedentes, el problema jurídico enunciado por la corte y la decisión tomada por la corte [1]. Esta estructura puede observarse en la figura 2.

Los antecedentes narran los hechos que han llevado a la vulneración de un derecho, quien ha sido el afectado y a quien se le solicita que el derecho restaurado. A partir los antecedentes el magistrado ponente identifica el problema jurídico, donde también se hace referencia a las normas que amparan el derecho vulnerado cuando es el caso y se han agotado correctamente las instancias anteriores. Por último, se encuentra la decisión donde se resuelve como se debe proceder, en el caso que la sentencia sea fallada a favor del demandante, o las razones por las cuales la tutela es fallada a favor del demandado.

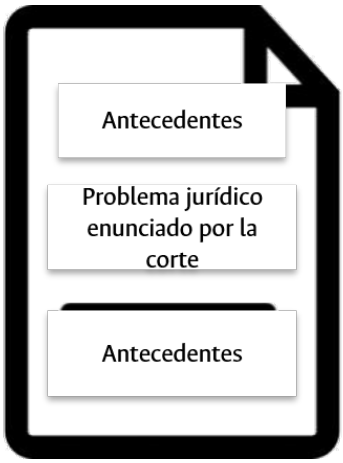

Figura 2: Estructura del documento

II-B. Crear un conjunto de datos objetivo

La generación del conjunto de datos base fue obtenido de la página de relatorías de la corte constitucional de Colombia[2], el contenido HTML fue tratados y posteriormente fue almacenado en MongoDb, la cual es una base de datos orientada a documentos. La base de datos tiene 8126 sentencias, donde cada sentencia cuenta con la fecha de sentencia, fecha de la recepción, el tipo de votación de los magistrados, el conteo de las palabras, el conteo de las entidades nominales, el texto de la sentencia y el texto del tema.

Con el objetivo de enfocarse en las sentencias donde se busca la restauración de derechos, el conjunto de datos para realizar el descubrimiento de conocimiento, se realizará sobre las sentencias tipo T, por lo cual al filtrar los registros, el conjunto de datos quedo compuesto de 3691 tutelas.

[2]https://www.corteconstitucional.gov.co/relatoria/

Para iniciar la exploración de datos, se evalúa la naturaleza de cada atributos y si existían datos en cada uno de los atributos más representativos, tal como se muestra a continuación

1. **date** : 3691 registros no nulos de tipo rango que representan la fecha de sentencia.
2. **sentence_id**: 3691 registros no nulos de tipo nominal que representan el identificador de la sentencia.
3. **topic**: 3640 registros no nulos de tipo nominal, el cual es un texto con el resumen de la sentencia
4. **report_receipt_at**: 3691 registros no nulos de tipo rango, que representan la fecha en la cual la relatoria es generada.
5. **text**: 3691 registros no nulos de tipo nominal que contiene el texto de la sentencia
6. **judicature**: 3691 registros no nulos de tipo nominal, el cual es un arreglo de datos donde se encuentra el nombre de los magistrados y tipo de voto.
7. **participants**: 3684 registros no nulos de tipo nominal, el cual es un arreglo de datos donde se encuentra el nombre de los participantes y el rol, donde pueden ser demandado o demandante.

Para los datos de rango, se identifica el rango en el cual oscilan las fechas, donde datos nulos en la fecha son representado por el valor de 1969-12-31, tal como se puede observar en el cuadro I. En la fecha de la relatoría se pueden observar datos nulos (report_receipt_at), dado que este dado es agregado posterior al fallo de la sentencia, este dado no será agregado al conjunto de datos objetivo.

	date	**report_receipt_at**
count	3691	3691
mean	2014-10-12 07:15:47	2015-01-16 15:19:33
min	2009-02-01 00:00:00	1969-12-31 00:00:00
25 %	2012-05-25 00:00:00	2012-10-05 00:00:00
50 %	2014-02-03 00:00:00	2015-02-11 00:00:00
75 %	2017-02-17 00:00:00	2017-05-05 00:00:00
max	2019-10-08 00:00:00	2020-03-15 00:00:00

Cuadro I: Descripción de datos tipo fecha

Al igual que la fecha de la relatoría datos como el *topic* (tema) no serán completados, pues son datos agregados y no todos los registros cuentan con este dato. Por lo tanto, para el conjunto de datos objetivo se usará la fecha de la sentencia (date), el identificador de la sentencia (sentence_id), el texto de la sentencia (text), los magistrados (judicature) y los participantes (participants), demandantes y demandados.

II-C. Limpieza y pre-procesamiento de datos

Para realizar el análisis exploratorio de las tutelas, los documentos fueron divididos en 2 secciones, una para los antecedentes y otra para las decisiones, esta división fue realizada a través de una expresión regular para obtener las secciones del documento, donde la primera sección, después de presentar las intensiones de la tutela son los antecedentes, y la quita sección refiere a las decisiones tomadas por el magistrado ponente.

Con el objetivo de analizar el texto de las sentencias, las palabras de los documentos fueron normalizados. Se realizaron dos procesos de normalización, donde se incluyen los siguientes pasos:

1. Eliminación de signos de puntuación
2. Eliminación de tildes
3. Conversión del texto a minúscula
4. Eliminación de palabras con menos de tres caracteres
5. Eliminación de números
6. Eliminación de stop words

Para la primera normalización, se ejecutaron todos los pasos con el objetivo de mejorar el proceso de presentación de los documentos eliminando palabras de alta repetición que no ayudan a la representación semántica como lo son las *stop words*.

Para la segunda normalización, tan solo se aplicaron los 3 primeros pasos, dado que se deseaba identificar las leyes y derechos relacionados, por lo que conservar los artículos determinados y números son importantes para identificar leyes como la ley 100 de 2002, compuesta de los elementos que son eliminados a partir de paso 4.

II-D. Reducción y proyección

El vocabulario de las sentencias, después de aplicar todos los pasos de la normalización, tiene un valor de $\mu = 1981$ palabras y $\sigma = 895$ palabras, por lo cual si se usarán métodos basado en el conteo de palabras, donde se use una cantidad de palabras que represente al 95 % de las sentencias, se tendrían que usar 3771 palabras, dado si se asume una distribución normal del vocabulario, ese valor es el resultado de $\mu + 2\sigma$.

Sin embargo, las representaciones de documentos basadas en conteo de palabras, aún si se aplicarán técnicas lematización o *stemming*, no mantiene la semántica de los documentos [3], y se generan representaciones *sparse* de alta dimensionalidad, por lo cuál, se usó una representación densa de las sentencias obtenida del método Doc2Vec.

Con el método de Doc2Vec cada palabra es mapeada a un único vector, representada por una columna en una matriz W. La columna es indexada por la posición de la palabra en el vocabulario. Además, cada sentencia es mapeada a un vector vector único, representado por una columna en la matriz D, donde, el identificador de la sentencia, puede ser pensado como otra palabra, la cuál actúa como memoria en el contexto, esto es llamado Distributed Memory Model of Paragraph Vectors (PV-DM).

El conjunto de palabras para representar cada documento es variable, sin embargo, la representación que es inferida tiene una dimensión fija. Para hallar el tamaño de la dimensión que mejor representa los documentos, se realizaron experimentos variando el espacio en potencias de 2, desde 2^2 hasta 2^10, donde se encontró que 128 es la dimensión que es óptima para representar un documento.

Para evaluar las dimensiones se usó una variación de la métrica *Normalized Discounted cumulative gain*(*nDCG*), donde solo se evalúa 1 *score*. Este proceso se aplica a cada uno de los documentos y se obtiene el promedio de cada uno de los valores.

$$nDCG = \frac{D_l - score}{D_l}$$

Para hallar el *score*, se infiere un vector a partir del texto procesado de la sentencia, luego se buscan las sentencias más similares entre el vector inferido y las sentencias usadas para entrenar el algoritmo de doc2vec, si la sentencia más similar obtenida es la misma del vector inferido, entonces el *score* de relevancia es 0. D_l, es la longitud de los documentos usados para evaluar la relevancia de las sentencias obtenidas.

A pesar que una ley o derecho sea nombrado múltiples veces en una sentencia, para las reglas de asociación tan solo se toman valores únicos, sin importar las veces que se repitan por sentencia, dado que se consideran como elementos únicos por documentos.

II-E. Empatar las metas a un método particular de minería de datos:

Para buscar algunas relaciones entre los derechos tutelados y las leyes que los protegen, se usaron reglas de asociación y algoritmos de agrupamiento. Para las reglas se asociación se usaron el algoritmo a Priori y el algoritmo de Eclat usando los derechos vulnerados como antecedentes y las leyes como consecuentes.

Los métodos de clustering ayudan a encontrar patrones entre documentos que sean similares usando la representación vectorial obtenida en la etapa anterior, sentencias similares deben agruparse por derechos que son vulnerados, de la misma forma, estos derechos deben invocar leyes similares que respalden la reclamación de la tutela.

Los algoritmos de clustering con los cuales se realizaron las pruebas son los algorimos de K-means, Gaussian mixure y DBScan, dado que estos métodos permiten evaluar la cercanía a los centroides basados en una medida de distancia como en el K-means y DBScan y con Gaussian mixure también se tiene en cuenta la dispersión de cada una de las dimensiones para evaluar esta cercanía.

II-F. Análisis exploratorio y selección de modelo e hipótesis:

Para la generación de reglas de asociación se realizó un proceso de transformación del texto, tal como se puede ver el la figura 3, donde solo se tomaron los atributos de texto, el cual se encuentra con el pre-procesamiento expuesto anteriormente, y la el id de la sentencia, luego el texto fue divido en el dos partes, en antecedentes y decisiones (la resolución de la sentencia), y de cada una de estas partes de obtuvieron el conjunto de referencias usadas como lo son artículos, decretos, leyes folios y otras sentencias. La unión de estos conjunto fue tratado como una transacción para la reglas de asociación.

Para generación de reglas de asociación se uso el algoritmo a Priori, pues este presentaba resultados similares al algoritmo de Eclat, sin embargo el algoritmo a Priori tuvo un mejor desempeño en tiempo procesamiento para el conjunto de datos usado.

Para evaluar las reglas de asociación se usó una confianza mínima de 0.8 y un soporte mínimo de 0.01. Para filtras reglas de mayor interese se filtraron relaciones entre los derechos como antecedentes y leyes como consecuentes, donde al menos dos leyes se encontraran relacionados con al menos un derecho con una confianza de 1 y un soporte de 0.034.

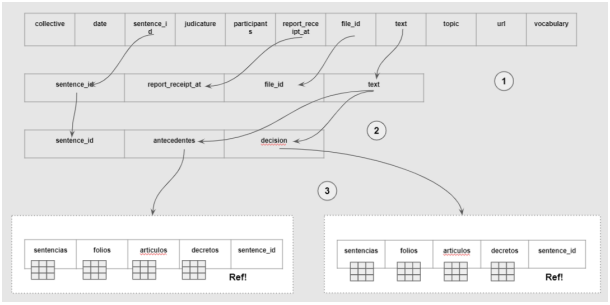

Figura 3: Proceso de transformación de texto a vectores de derechos y normas

Para evaluar los algoritmos de agrupamiento y determinar el mejor número de grupos, se evaluó el coeficiente de silueta usando como métrica de la distancia de coseno. La cantidad de grupos probados fue desde 2 grupos hasta 30 grupos como se puede ver en la figura 4 y en la figura 5.

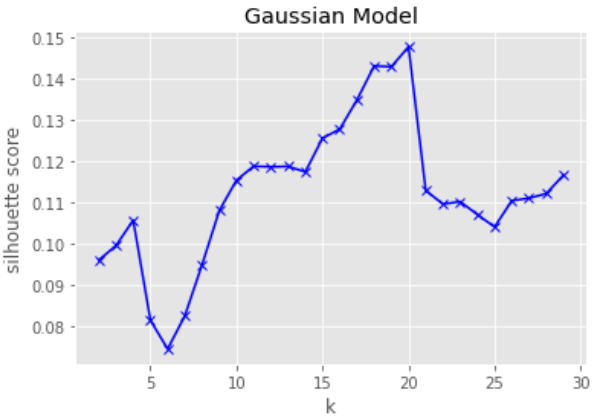

Figura 4: Evaluación del número de grupos del modelo Gaussian Mixture usando el coeficiente de silueta

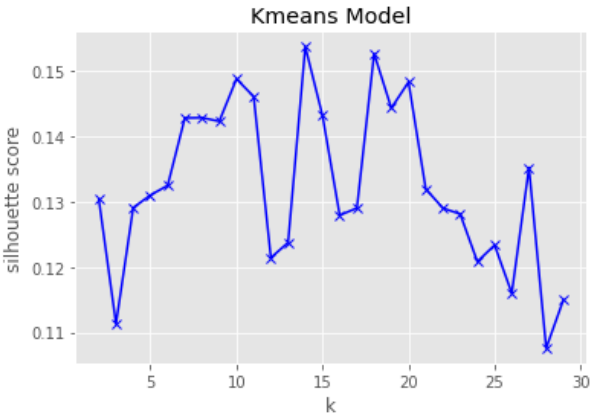

Figura 5: Evaluación del número de grupos del modelo K-means usando el coeficiente de silueta

De los experimentos realizados, usando el algoritmo de *k-means* para representar los diferentes grupos de sentencias, se encontró que 14 grupos es la cantidad óptima para este

Modelo	Parámetros	Ensamble	n estimadores	F1
SVC	kernel: rbf			0,925
Gaussian		boosting	30	0,898
KNN	n_neighbors:5	boosting	30	0,898
Gaussian		boosting	40	0,895
KNN	n_neighbors:5	boosting	40	0,895
KNN	n_neighbors:5	boosting	20	0,895
SVC	kernel: cosine			0,894
Gaussian		boosting	50	0,894
KNN	n_neighbors:5	boosting	10	0,888
SVC	kernel: lineal	bagging	30	0,888

Cuadro II: Resultado de 10 mejore modelos

objetivo, obteniendo un valor de superior a 0.15 con el coeficiente de silueta.

Con los resultados del algoritmo de agrupamiento se realizo un análisis del texto de cada uno de los grupos, y a cada cluster se le asignaron los principales temas que son tratados. En la figura 6, se presenta la distribución de los temas, donde se puede evidenciar que se trata con un problema de imbalance de clases.

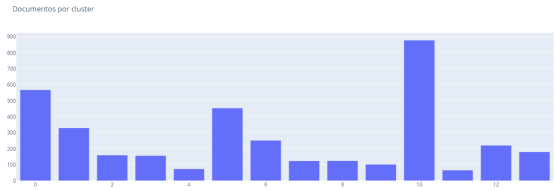

Figura 6: Distribución de temas de los documentos

Para realizar las pruebas el conjunto de datos fue inicialmente participando 80 % de los datos para entrenamiento y 20 % para pruebas.La partición de los datos se realizó a través de un muestro estratificado con el objetivo de mantener la proporción de los temas.

Para la selección del modelo se probaron más de 60 modelos donde se realizaban cambios en los parametros de modelos como Support Vector Clasification (SVC), k-nearest neighbors (KNN), Gaussian Naive Bayes, arboles de decisión y modelos de emsamble como Boosting, Bagging y Random o Forest usando un optimizador bayesiano [3] que seleccionará los mejores parámetros que optimizaban la medida f1 a traves de una validación cruzada de 5-Folds, partiendo los datos en 5 carpetas, donde una se dejaba para pruebas y las 4 restantes para entrenamiento.

Dentro de los parámetros a optimizar el tipo de modelo que se usaría se contemplo, también el hecho de si ese modelo debía ser usado como modelo de ensamble y el tipo de modelo de ensamble que debía usarse y la cantidad de estimadores para el modelo de ensamble. En la tabla II, se muestra el modelo con mejor resultado fue el método de SVC usando un kernel de función radial.

De los resultado de la clasificación se obtieron los siguientes resultados en cuando al desempeño del mejor algorimo seleccionado como se puede ver en la tabla III

[3]https://keras.io/api/keras_tuner/tuners/bayesian/

	precision	recall	f1-score	support
acto administrativo	1.00	0.97	0.98	31
comunidades indigenas	1.00	0.96	0.98	25
debido proceso	0.95	0.91	0.93	247
desplazamiento	1.00	0.93	0.96	68
educacion	1.00	0.86	0.93	44
estabilidad laboral	1.00	0.92	0.96	51
establecimiento penitenciario	1.00	1.00	1.00	20
libertad de expresion	1.00	0.80	0.89	15
minimo vital	0.96	0.94	0.95	303
pension	0.96	0.99	0.97	176
providencias judiciales	0.98	0.95	0.97	66
reparacion integral	1.00	0.91	0.95	32
seguridad social	0.97	0.99	0.98	432
seguro	1.00	1.00	1.00	13
servicio publico	1.00	0.88	0.94	25
servicios de salud	0.99	0.99	0.99	91
spe	1.00	0.86	0.93	44
vivienda digna	1.00	0.94	0.97	36

Cuadro III: Resultado de la clasificación

II-G. Minería de datos:

Para observar la calidad de los grupos obtenidos, se visualizan por medio de una representación bidimensional del vector de documentos normalizado, usando el algoritmo de TSNE[4], como se muestra en la figura 7

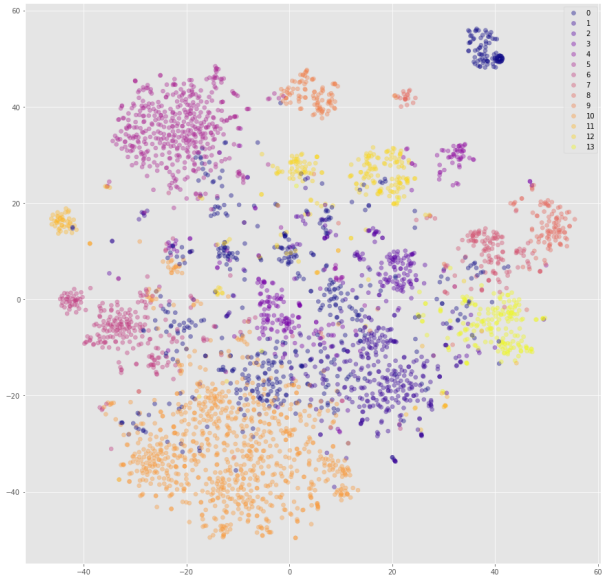

Figura 7: Visualización de grupos encontrados

Se puede observar que los grupos grandes se encuentra más dispersos, y algunos grupos pequeños se encuentran más condensados, lo cual disminuye el coeficiente de silueta. Además, algunas sentencias se puede observar que se encuentran solapadas, por tanto, tumben se podría considerar el hecho que una sentencia pueda pertenecer a varios grupos.

Después de aplicar el algoritmo a *Priori*, dado su alto nivel de confianza y la cantidad de ejemplos que lo soportan, las 3 reglas más importantes que se encontraron fueron las siguientes, :

Regla 1: 'derechos fundamentales a la salud' → {'decreto 806 de 1998', 'ley 10 de 1990'}

[4]https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

Regla 2: 'derechos fundamentales a la seguridad' → {'decreto 691 de 1994', 'ley 100 de 1993'}

Regla 3: 'derechos fundamentales al mínimo vital' → {'decreto 3770 de 2008', 'ley 70 de 1993'}

II-H. Interpretación de patrones minados:

Para interpretar los grupos obtenidos, se genera una nube de palabras, como se puede ver en la figura 8, donde se omiten palabras de alta frecuencia que son comunes en el contexto de las tutelas como derecho, fundamental, sentencia, Colombia y magistrado.

Figura 8: Nube de palabras

De las palabras de cada uno de los grupos, se puede observar que aquellas que más resaltan se encuentran relacionas con los derechos que están siendo vulnerados y los eventos que llevaron a ese hecho. En algunos grupos, cuando se trata de derechos colectivos lo que más resalta es la comunidad que sufre de la vulneración de estos derechos, como en el caso de comunidades indígenas o desplazados.

La generación de reglas de asociación permite observar que los derechos mayormente tutelados son los derechos a la salud, la seguridad social y al mínimo vital, lo cuál coincide con los cluster 0, cluster 5,cluster 6 y cluster 10, donde al observar las palabras frecuentes se pueden ver las entidades y eventos que generaron la tutela.

III. Resultados

El algoritmo de clustering que mejor agrupa las sentencias es el algoritmo de k-means. Para mejorar el rendimiento de los algoritmo de agrupación el vector denso obtenido del Doc2Vec fue normalizado. El proceso realizado para obtener los resultados de agrupación se encuentran disponibles en un notebook publicado en GitHub[5].

Las reglas de asociación permiten observar los derechos que son mayormente tutelados y las leyes sobre las cuales se fundamenta la tutela, como se puede observar en el notebook publicado en el repositorio[6], estos resultado son similares a los obtenidos con los grupos obtenidos, donde las leyes que no se pueden observar en los grupos se pueden complementar con estas reglas.

El entrenamiento del modelo Doc2Vec con las sentencias de la corte permitió generar un modelo que puede inferir una representación vectorial de baja dimensionalidad, usando solo 128 valores en un vector denso, con la capacidad de mantener la semántica de las sentencias en el contexto, por lo tanto también es posible identificar sentencias simulares dado un conjunto de palabras usadas.

IV. Conclusiones

La representación de las sentencias, por medio de un embedding denso obtenido del uso del método de Doc2Vec, permite mantener la semántica y la similitud entre las sentencias y luego usar algoritmos clásicos como el K-means para realizar tareas especificas, como fue la identificación de grupos

A pesar que el algoritmo de clustering no permitió identificar la relación entre derechos y leyes, si permite identificar la relación entre derechos y principales eventos e identidades relacionadas, lo cuál es importante para relacionar sentencias con antecedentes similares.

Para lograr las metas del proceso, los resultados de un solo método pueden ser complementados con otros métodos para obtener los resultados esperados, generando artefactos que pueden superar las metas inicialmente planteadas que resuelven problemas que se presentan en el dominio del negocio, como el poder encontrar sentencias relacionadas tan solo con algunas palabras de intereses, como aquellas que hacen parte de los antecedentes.

La relación entre los derechos vulnerados y las leyes que amparan estos derechos puede guiar a las personas no expertas en leyes a conocer las garantías que ofrece el estado.

[5]https://github.com/leuder/mining_judgments/blob/master/Proyecto_-Agrupación.ipynb

[6]https://github.com/leuder/mining_judgments/blob/master/entrega_reglas.ipynb

Identificar casos que tengan hechos y pretensiones similares a través del análisis del texto permitiría agruparlos de forma automática con el objetivo de mejorar el proceso que se realiza manualmente.

Identificar casos relacionados con providencias judiciales desde las primeras instancias y la jurisprudencia relacionada con respecto a la interpretación de las leyes, ayudaría que menos casos llegarán a la corte suprema por falta al debido proceso o mala interpretación del principio de subsidiariedad.

V. Trabajo futuro

Al realizar la extracción de normas que soportan el razonamiento jurídico, se evidencia que si bien algunas referencias legales son fácilmente identificables por medio de expresiones de regulares, otras normas y referencias no siguen un patrón fácilmente identificable y generalizable, por lo tanto, se podrían entrenar algoritmos de aprendizaje de máquina para reconocer las normas como entidades legales, y así mejorar el análisis de asociación entre los derechos vulnerados y las normas que los protegen, sin usar expresiones regulares de alta complejidad y difícil mantenimiento.

Referencias

[1] Jalil Alejandro Magaldi Serna. Propuesta metodológica para el análisis de sentencias de la corte constitucional. *Serie Documentos de Trabajo, Departamento de Derecho Constitucional*, 2014.

[2] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. Knowledge discovery and data mining: Towards a unifying framework. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 82–88. AAAI Press, 1996.

[3] Quoc V. Le and Tomás Mikolov. Distributed representations of sentences and documents. *CoRR*, abs/1405.4053, 2014.

[4] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.