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

Clasificación Multinivel de Fallos Judiciales para una Editorial Legal

Mauro J. Pacchiotti[1], Milagros Gutierrez[1], Mariel Ale[1]
[1]Centro de I+D de Ingeniería en Sistemas de Información UTN-FRSF
{mpacchiotti, mgutierrez, male}@frsf.utn.edu.ar

Resumen

En el ámbito legal, los profesionales se encuentran recurrentemente con la necesidad de encontrar fallos relacionados a la temática que está tratando. La editorial jurídica con la que se trabajó ofrece el servicio de búsqueda y recuperación de dichos fallos. Para hacerlo necesita clasificar los mismos de acuerdo con un tesauro propio, consistente de tres niveles, que representan los temas de interés jurídico. La automatización de esta clasificación, a través del uso de técnicas de inteligencia artificial, permitió acelerar los tiempos de disponibilidad de los fallos y por ende el servicio brindado a sus clientes. En este trabajo se presenta la solución que se brindó para clasificar más de 200.000 fallos en múltiples categorías, para lo cual se realizó un trabajo de análisis y depuración del dataset como también selección y entrenamiento de modelos.

1. Introducción

La clasificación de textos ha tenido un desarrollo vertiginoso en los últimos años. Esto se debe, entre otras cosas, a la disponibilidad de texto digitalizado, como páginas web, correos electrónicos, blogs, bibliotecas digitales, anuncios en línea, documentos corporativos, reseñas de productos, entre otros. Muchas aplicaciones basadas en estas diferentes fuentes de datos pueden plantearse como problemas de clasificación de texto. En estos problemas, es necesario clasificar los documentos en clases predefinidas que representan diferentes grupos semánticos (por ejemplo, spam y no spam, tópicos o sentimientos).

Independientemente de los avances, la clasificación de textos sigue presentando un conjunto de desafíos a resolver. En primer lugar, los documentos de texto están re-presentados de una forma dispersa en un espacio de términos de muy alta dimensión, lo que dificulta el aprendizaje y la generalización. En segundo lugar, debido al elevado coste del etiquetado de los documentos, los investigadores se ven obligados a recurrir a pequeños conjuntos de entrenamiento o a recopilar datos de entrenamiento procedentes de fuentes distintas del dominio de destino. Esto da lugar a un cambio de distribución entre los datos de entrenamiento y los de prueba. En tercer lugar, los documentos son de diferente calidad, idioma y longitud, lo que hace que un enfoque uniforme basado en el conocimiento sea ineficaz o inviable [1].

En la actualidad la clasificación de documentos con etiquetas múltiples tiene una gran variedad de aplicaciones en el mundo real que van desde, la clasificación de publicaciones en redes sociales [2], el análisis de sentimientos [3], hasta la clasificación de códigos de diagnósticos médicos [4]. Se han desarrollado diversos métodos para este tipo de problemas, entre los que se incluyen los clasificadores tradicionales one vs. all [5,6], los enfoques clásicos de aprendizaje automático (por ejemplo, Random Forest [7] y Perceptron Multicapa [8]) y las Redes Neuronales Profundas [9][10][11]. Los avances en el aprendizaje automático y otros campos relacionados han permitido que los algoritmos de clasificación multietiqueta logren mejoras continuas en conjuntos de datos de diferentes dominios [12][13][14].

A nivel internacional y dentro del ámbito jurídico específicamente, existe una fuerte demanda de algoritmos de clasificación multietiqueta de alto rendimiento para diferentes tareas, como la detección de mociones y órdenes [15] y la predicción de resultados de casos [16]. Sin embargo, los investigadores y los profesionales se enfrentan a menudo a dos grandes retos. Por un lado, sólo existen unos pocos conjuntos de datos textuales legales anotados por humanos [17,18], y la falta de datos etiquetados manualmente de alta calidad se ha convertido en un gran obstáculo para seguir avanzando en la investigación de vanguardia en este campo. Por otra parte, aunque los métodos existentes han conseguido rendimientos aceptables en diversas tareas, se centran principalmente en las clases mayoritarias y tienen dificultades para conseguir un rendimiento decente para las clases que no tienen suficientes muestras de entrenamiento. Por ejemplo, para la clasificación de fallos judiciales en ramas del derecho, existen clases menos frecuentes que otras. Así, por ejemplo, en la rama Civil y Comercial hay una gran cantidad de fallos, mientras que en la rama Penal Comercial hay menos. La omisión de estas categorías poco frecuentes, pero importantes, puede tener consecuencias importantes para las tareas de clasificación en este contexto.

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

En este trabajo se presenta el resultado de clasificar 240.181 fallos de acuerdo con un tesauro de 3 niveles, donde en el primer nivel existen 8 clases, en el segundo nivel 931 clases y en el último nivel 3.459. Tanto el dataset como el tesauro recibido están protegidos por un acuerdo de confidencialidad ya que pertenece a una editorial legal que comercializa su acceso. Los fallos judiciales constituyen una fuente muy importante de jurisprudencia para los profesionales judiciales y la empresa editorial busca ofrecer un sistema de búsqueda y recuperación que exceda la simple búsqueda de palabras en el texto clasificando los fallos por temática y, de esta forma, mejorar los resultados presentados a sus clientes. En este sentido se planteó una estrategia que permitiera sortear las principales dificultades encontradas: la gran cantidad de etiquetas disponibles y la ausencia de ejemplos para muchas de ellas.

El resto del artículo está organizado de la siguiente manera. En la sección 2 se analizan los trabajos relacionados con el tema. Las secciones 3 y 4 presentan los detalles del conjunto de datos y la estrategia de clasificación propuesta, respectivamente. En la sección 5 evaluamos el sistema propuesto y en la sección 6 se presentan conclusiones y trabajos futuros.

2. Trabajos Relacionados

La clasificación de documentos legales consiste en identificar la categoría de un texto jurídico basándose en la asociación entre el texto jurídico y esa categoría. Existen una variedad de tareas que están relacionadas al proceso de clasificación: la categorización en áreas del derecho, la identificación de sentencias, la minería de argumentos y la predicción de decisiones judiciales. En los últimos años se han realizado muchos estudios sobre la clasificación de textos legales. Por ejemplo, Palau y Moens [19] identificaron las proposiciones argumentativas, la función y la estructura argumentativas en textos legales del Tribunal Europeo de Derechos Humanos. Boella y col. [20] clasificaron textos jurídicos en italiano en un dominio relevante. Aletras y col. [21] también trabajaron sobre documentos del Tribunal Europeo de Derechos Humanos intentando predecir la sentencia, el área jurídica y la fecha de emisión de la sentencia. Sulea y col. [22,23] aplicaron técnicas de aprendizaje automático para predecir sentencias del Tribunal Supremo francés y el área jurídica a la que pertenece un caso. Ji, Tao, y col. [24] incorporaron la tarea de clasificación legal a la tarea de extracción de información como un problema de aprendizaje multitarea para la extracción de pruebas de documentos judiciales chinos. Posteriormente, aplicaron los mismos textos legales para la resolución de coreferencias de hablantes [25].

Luz de Araujo y col. [26] presentan los resultados de aplicar una diversidad de en-foques (bag-of-words, redes convolucionales, redes recurrentes y algoritmos de boosting) a un conjunto de documentos legales digitalizados pertenecientes a la Corte Suprema de Justicia de Brasil para realizar clasificación y asignación de temas. Otro trabajo en textos en portugués es el presentado por Domingues y col. [27] quienes utilizan una estrategia de clasificación para apoyar el análisis de los documentos jurídicos que citan o podrían citar un precedente vinculante creado por la Corte Suprema de Justicia.

Li [28] desarrolla una estrategia de clasificación de documentos legales basada en la extracción de palabras características representativas de cada documento. Dichas palabras son determinadas a través de TF-IDF y luego modificadas con un factor de corrección que tiene en cuenta la posición de la palabra en el texto.

Song y col. [29] presentan un conjunto de datos de aproximadamente 50.000 opiniones legales (POSTURE50K) y proponen una arquitectura de aprendizaje profundo que adopta un preentrenamiento específico del dominio y un mecanismo de atención a la etiqueta para la clasificación de documentos.

Chen y col. [30] presentan un algoritmo de aprendizaje automático que utiliza conceptos de dominio como características y bosques aleatorios como clasificador de textos jurídicos con una gran colección de documentos de casos de EE.UU. (SigmaLaw) etiquetados en 50 categorías.

Priyadarshini y col. [31] proponen un método de recuperación de información semántica que pretende ir más allá de la recuperación de información estándar que permitiría obtener documentos jurídicos relacionados dentro de un corpus. El sistema propuesto identifica los documentos legales y aumenta la precisión y el rendimiento del análisis de los mismos utilizando un modelo ensamblado.

Finalmente, y a nivel local, se encuentra la propuesta de Perezzini, Casali y Deco [32] que presentan un sistema de soporte para recuperar de forma periódica y semi-automática, normativas potencialmente relevantes con respecto a las actividades realizadas por una empresa. Dicho sistema realiza un proceso de clasificación de normativas utilizando clasificadores binarios con el corpus y las categorías presentes en el Sistema Argentino de Información Jurídica (SAIJ).

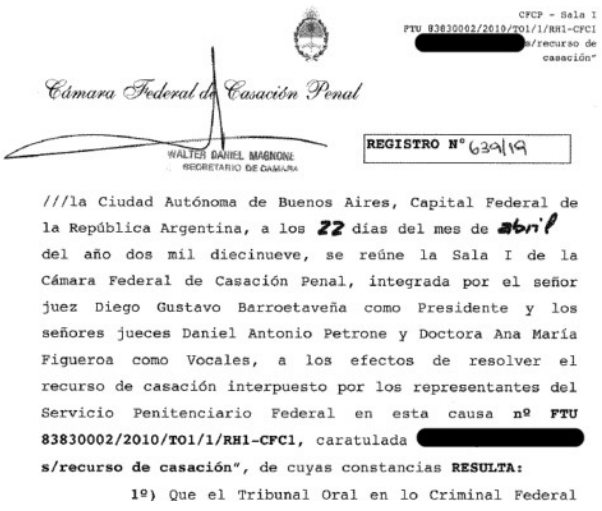
Si bien se observa una variedad de técnicas aplicadas para la clasificación de documentos legales con resultados disímiles la particularidad de la tarea a resolver para la editorial legal tanto en el corpus de documentos (normalmente los fallos son de mayor longitud y heterogéneos en cuanto a su estructura y contenido) como en el esquema de etiquetas multinivel propio de la empresa, conducen a la necesidad de repensar un enfoque diferente que se describe en las secciones siguientes.

3. Conjunto de Datos

El conjunto de datos consiste en 240181 fallos judiciales en diferentes formatos (en su mayoría archivos PDF) que pueden contener 1 o más sumarios relacionados

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

a diferentes categorías y subcategorías del derecho (Figura 1). El primer desafío fue extraer el texto a partir de los archivos PDF. En esa instancia también se realizó, a pedido de la editorial, un proceso de extracción de características utilizando expresiones regulares que permitían identificar el número de fallo, la carátula, las partes involucradas, la fecha, el tribunal y el texto del fallo. Una parte de los datos recibidos estaban constituidos por imágenes (documentos escaneados) que debieron pasar un proceso previo de reconocimiento de caracteres.



CFCP - Sala I
FTU 83830002/2010/TO1/1/RH1-CFC1 [redacted] s/recurso de casación"

Cámara Federal de Casación Penal

WALTER DANIEL MAGNONI
SECRETARIO DE CAMARA

REGISTRO N° 639/19

///la Ciudad Autónoma de Buenos Aires, Capital Federal de la República Argentina, a los 22 días del mes de abril del año dos mil diecinueve, se reúne la Sala I de la Cámara Federal de Casación Penal, integrada por el señor juez Diego Gustavo Barroetaveña como Presidente y los señores jueces Daniel Antonio Petrone y Doctora Ana María Figueroa como Vocales, a los efectos de resolver el recurso de casación interpuesto por los representantes del Servicio Penitenciario Federal en esta causa nº FTU 83830002/2010/TO1/1/RH1-CFC1, caratulada [redacted] s/recurso de casación", de cuyas constancias RESULTA:

1º) Que el Tribunal Oral en lo Criminal Federal

Figura 1. Ejemplo de fallo judicial.

Como resultado del proceso de extracción de características se obtuvieron 240.181 archivos en formato JSON. Cada uno de estos archivos contenía los datos y texto de un fallo judicial y uno o más sumarios relacionados. A pedido de la empresa la clasificación debía realizarse a nivel de sumario. Los sumarios son porciones del texto del fallo que tratan sobre algún tema particular de interés y que son seleccionados y extraídos por profesionales del derecho denominados sumariantes. Cada uno de los sumarios tiene una o más categorías asociadas que son determinadas por los sumariantes. Esto introdujo un nuevo desafío en la problemática ya que se estaba en presencia de un esquema de etiquetado multinivel.

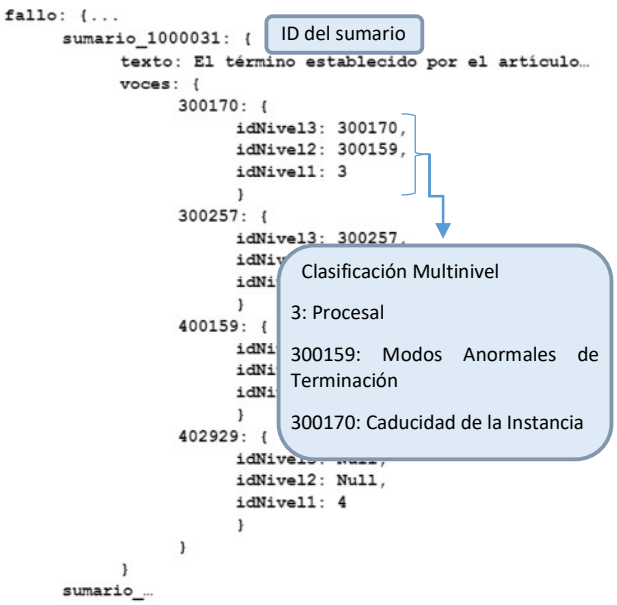


Figura 2. Ejemplo de etiquetado de un sumario de un fallo.

Cada uno de los sumarios (asociados a su correspondiente fallo) contenidos en los archivos JSON fue etiquetado por el sumariante que lo confeccionó. La estructura final puede observarse en la Figura 2.

3.1 Conjunto de datos para pruebas iniciales

Inicialmente y a los fines de probar distintas estrategias de preprocesamiento y modelos de clasificación se confeccionó, a partir de los archivos JSON, un primer dataset en el que cada muestra constó del texto del sumario y la etiqueta (denominada por la empresa como "voz") correspondiente al Nivel 1 de clasificación (Figura 3).

{'texto' : texto_del_sumario, 'voz' : voz_de_nivel_1}

Figura 3. Estructura de datos del dataset utilizado en las pruebas iniciales.

Sobre este dataset inicial se realizó un análisis exploratorio que mostró un marcado desbalance entre la cantidad de muestras de las categorías a clasificar (Figura 4). Las etiquetas del nivel 1 contenían, de origen, una numeración del 2 al 9.

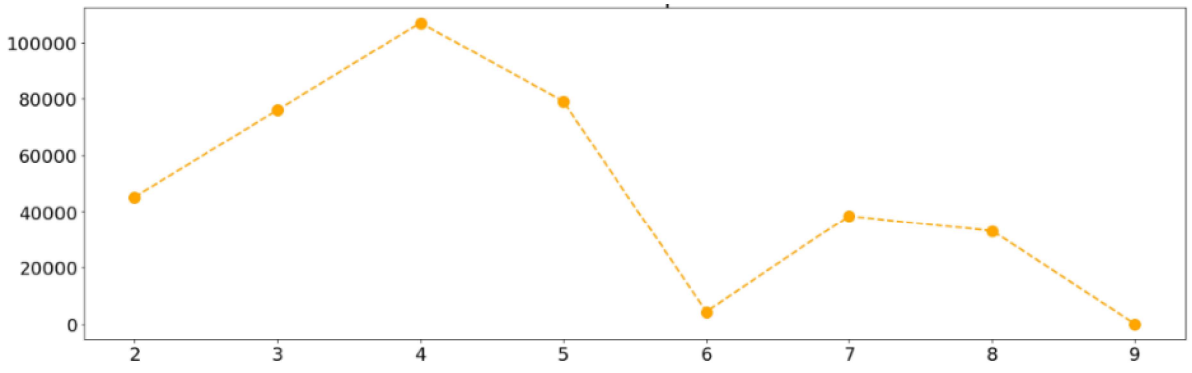


Figura 4. Cantidad de sumarios por categoría de Nivel 1.

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

A continuación, se realizaron los primeros procesos sobre este conjunto de datos (Figura 5) que comprendieron las siguientes operaciones: pasaje a minúsculas, eliminación de caracteres especiales, *stop words* y palabras de longitud menor a tres caracteres. Todas estas tareas tuvieron como objetivo normalizar los textos y eliminar palabras que no aporten a la solución del problema de clasificación.

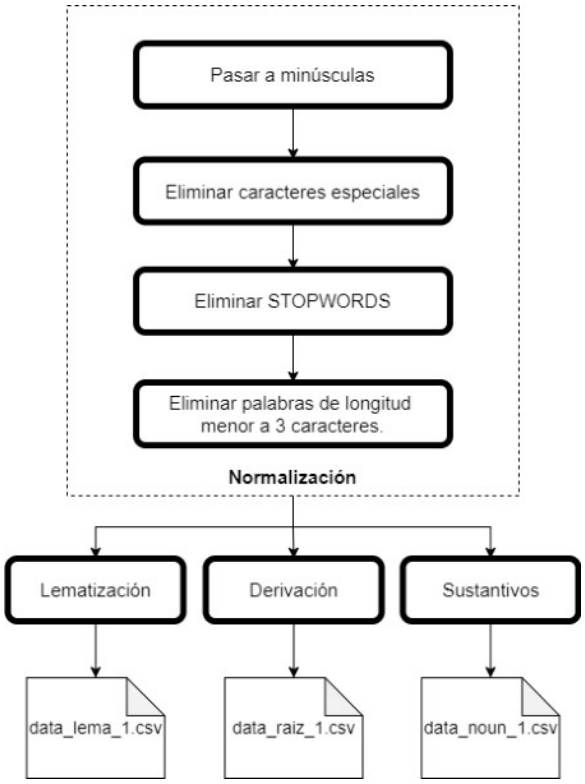


Figura 5. Diagrama de flujo del proceso de normalización

A partir de este conjunto de datos normalizado, se realizaron tres copias a las que se aplicaron distintas técnicas, según se describe a continuación: lematización [33] al primero, derivación [33] al segundo y selección solo de sustantivos al tercero, finalizando esta etapa con la creación de tres archivos que contenían los resultados de cada proceso a fin de que sirvieran como entrada a la etapa siguiente.

Una vez obtenidos los tres conjuntos de datos normalizados, se aplicó una secuencia de procesos con el fin de terminar el preprocesamiento de los textos, vectorizando y optimizando su representación Este preprocesamiento consistió en aplicar la técnica de bolsa de palabras [34] para vectorizar los textos y a estos vectores se los procesó utilizando TF-IDF [35] de esta manera se optimizan los textos resaltando las palabras que más los caracterizaban. En la Figura 6 puede verse un diagrama de flujo con los procesos aplicados en esta última etapa.

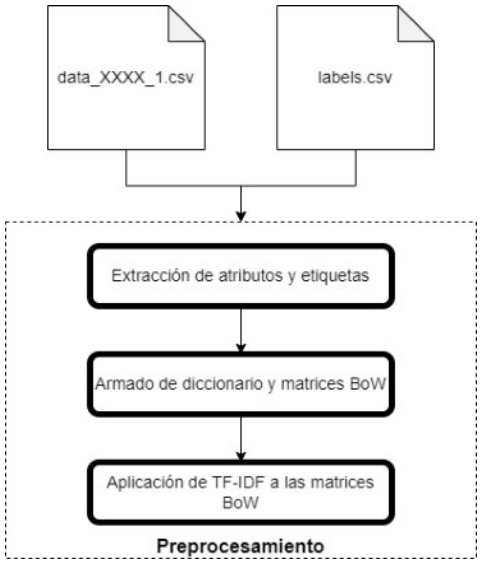


Figura 6. Diagrama de flujo del preprocesamiento.

El objetivo final del armado de estos 3 dataset (con las clasificaciones sólo a nivel 1) fue probar diferentes estrategias de pre-procesamiento seguido de su clasificación, también con distintos modelos.

4. Estrategias de Clasificación

4.1 Pruebas iniciales de modelos

Con los tres conjuntos de datos para pruebas iniciales se entrenaron dos modelos: un KNN [36] y un Regresor Logístico [37] para clasificar sumarios en la categoría de primer nivel correspondiente. La selección de estos modelos se basó en la pequeña cantidad de muestras presentes en las clases minoritarias. Si bien el conjunto de datos presenta un marcado desbalance en la cantidad de muestras por categoría a clasificar, este se utilizó en esta primera instancia de pruebas solo con fines comparativos, para medir el desempeño de distintas técnicas de preprocesamiento y modelos de clasificación. Las muestras se dividieron en dos conjuntos, uno para entrenamiento con el 70% y uno para test con el 30% restante, salvo en las categorías 6 y 9 donde las muestras no eran suficientes para realizar la división del conjunto y, a pesar de no ser recomendable, se utilizó el total de estas tanto para el entrenamiento de los modelos, como para la etapa de prueba.

$$Accuracy = \frac{Cantidad\ de\ Predicciones\ correctas}{Cantidad\ total\ de\ predicciones} * 100$$

Figura 7. Cálculo de Accuracy

La Tabla I presenta los resultados para cada conjunto de datos obtenido con las distintas técnicas de preprocesamiento y cada modelo, para medir los desempeños se utilizó el Accuracy (Figura 7). Como se observa, el mejor resultado se obtuvo con el regresor logístico en todos los casos. Basados en estos resultados preliminares, se decidió continuar los trabajos utilizando

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

Lematización como técnica de pre-procesamiento y el Regresor Logístico como modelo de clasificación.

Tabla 1. Precisión obtenida por cada combinación de tipo de preprocesamiento y modelo.

Preprocesamiento	Modelo	Accuracy
Lematización	**KNN**	**68,67%**
	Regresor Logístico	**70,97%**
Derivación	**KNN**	**68,74%**
	Regresor Logístico	**70,98%**
Solo sustantivos	**KNN**	**64,31%**
	Regresor Logístico	**67,89%**

4.2 Estrategia seleccionada para el problema de clasificación

Habiendo seleccionado la secuencia de preprocesamiento se prepara finalmente, a partir de los archivos JSON, el dataset final donde cada muestra constaba del texto del sumario y las voces de clasificación de Nivel 1 y Nivel 2. En esta primera etapa, la empresa consideró no incluir el nivel 3 por la escasez de ejemplos para muchas de las categorías. Los textos de los sumarios se procesaron entonces en esta secuencia: pasaje a minúsculas, eliminación de caracteres especiales, eliminación de *stop words*, eliminación de palabras de longitud menor a 3 caracteres, vectorización mediante Bolsa de Palabras y aplicación de TF-IDF a los vectores resultantes. Luego de este preprocesamiento todos los sumarios quedan representados con un vector la misma longitud.

De acuerdo con la estrategia de clasificación elegida, fue necesario generar un dataset por cada categoría de Nivel 1 y por cada categoría de Nivel 2. Para balancear cada uno de estos datasets se optó por incorporar al mismo los casos positivos de la clase en cuestión y la misma cantidad de casos negativos seleccionados al azar entre las restantes muestras. Por último, previo al entrenamiento del modelo, cada conjunto de datos se dividió en un 70% para el proceso de entrenamiento y un 30% para test posterior.

Con los resultados de los análisis realizados al conjunto de datos y los resultados obtenidos en las pruebas iniciales se llegó a varias conclusiones que desafiaron la aplicación del modelo a utilizar. En resumen, se abordó un problema de clasificación multinivel, el conjunto de datos mostraba un marcado desbalance en las cantidades de muestras por categoría, las categorías minoritarias poseían muy pocas muestras (menos de 5 en algunos casos) por último, la clasificación era multi-categoría, esto quiere decir que un sumario podía corresponderse con más de una categoría en el mismo nivel.

Con un problema de estas características se planteó como solución descomponerlo en problemas binomiales, entrenando un modelo de Regresor Logístico para cada categoría de nivel 1 y cada categoría de nivel 2.

Luego se implementó un prototipo (Figura 8) que, dado

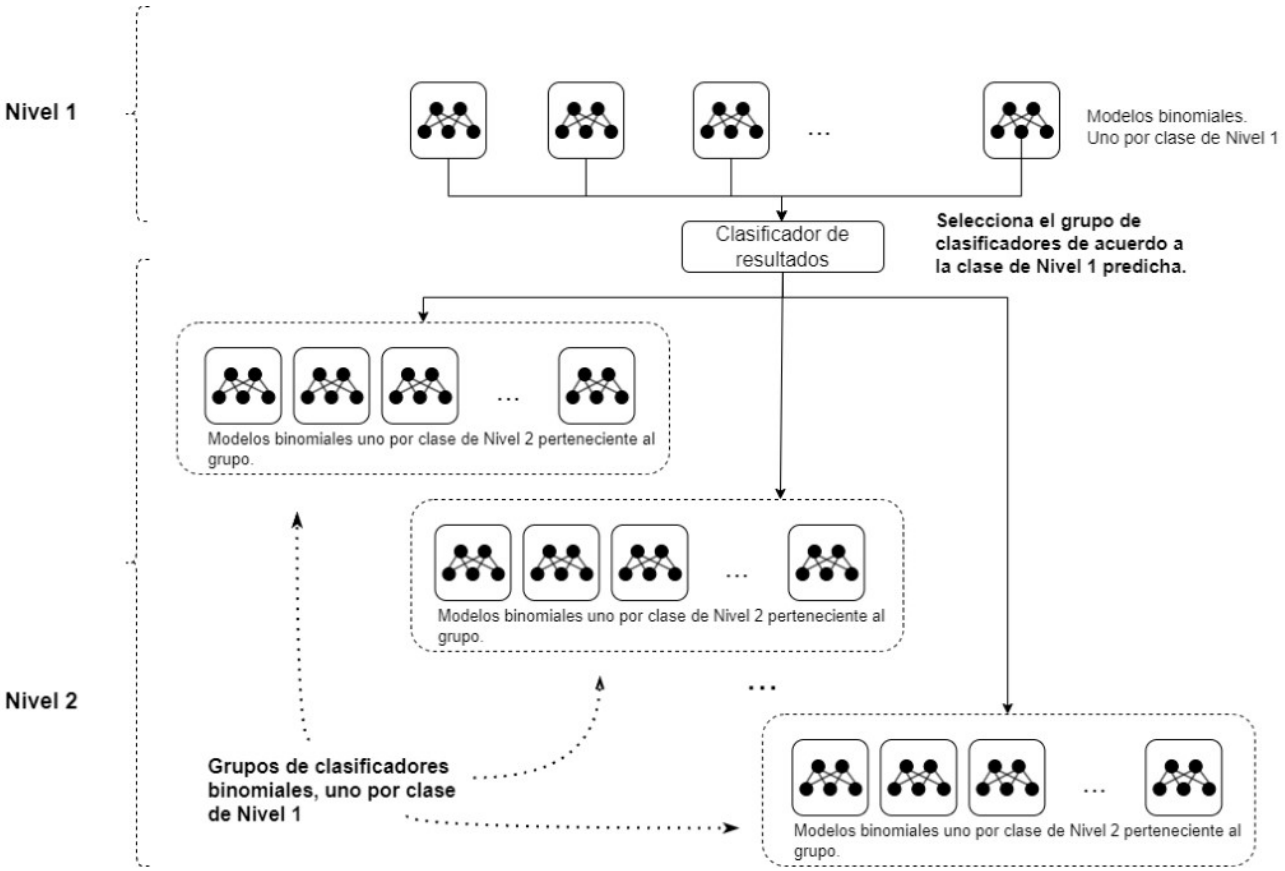


Figura 8. Diagrama de flujo del proceso de clasificación.

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

un texto de sumario a clasificar, primero lo haga con los 8 modelos para el Nivel 1. Una vez obtenidas las categorías predichas de Nivel 1, se seleccionan los modelos de las categorías de Nivel 2 correspondientes a los de Nivel 1 predichos y se realiza una nueva clasificación del texto del sumario, obteniendo de esta forma las categorías predichas de Nivel 1 y Nivel 2, pudiendo ser estas múltiples en ambos niveles.

Debido al gran volumen de datos y la gran cantidad de artefactos que se generarían mediante esta estrategia, se automatizó el proceso de generación de todos los artefactos necesarios para la implementación de la solución mediante un pipeline de procesos.

La primera etapa procesó los fallos en archivos individuales para obtener un único archivo JSON con un sumario por registro con la correspondiente etiqueta de Nivel 1 y Nivel 2 y el texto pre-procesado (pasado a minúsculas y sin caracteres especiales, *stop words* o palabras de longitud menor a tres caracteres).

La segunda etapa divide y replica el archivo generado en la primera para obtener los conjuntos de datos de primer y segundo nivel, generando un archivo CSV de sumarios y uno de etiquetas por cada clase de cada nivel.

Por último, la tercera etapa toma los archivos generados en la anterior y por cada categoría a clasificar evalúa, si hay menos de 6 muestras en el conjunto de datos, no prepara un clasificador para esa categoría, caso contrario vectoriza los textos mediante la técnica de bolsa de palabras, para luego mejorar la representación utilizando la transformación TF-IDF. A partir de los textos vectorizados y procesados y los archivos de etiquetas, se genera un modelo de Regresor Logístico por cada conjunto de datos (cada categoría de cada nivel) y se entrena. Esta etapa devuelve cuatro archivos: el diccionario utilizado para vectorizar los textos, el transformador TD-IDF entrenado con los textos del entrenamiento, el modelo de regresión logística entrenado y un archivo CSV con los resultados del entrenamiento.

4.3 Modelo y parámetros utilizados

Como se cita en la sección anterior, cada uno de los modelos constó de un Regresor Logístico y el ajuste de los hiperparámetros (Tabla 2) se hizo de acuerdo con una configuración recomendada para la librería utilizada, Scikit-Learn [38], en conjuntos de datos de gran dimensión.

Tabla 2. Hiperparámetros aplicados al Regresor Logístico.

Hiperparámetro	Valor
Solver (Método de entrenamiento)	**Liblinear**
Penalty (Método de regularización)	**L2**
Max_iter (Máximo de iteraciones para el entrenamiento)	**1000**

5. Evaluación

La evaluación de la estrategia seleccionada no puede realizarse con una única medida debido a la gran cantidad de modelos resultantes para las diferentes etapas. Luego de la ejecución del prototipo se generaron 8 modelos para la clasificación en nivel 1 y 427 modelos para la de nivel 2, el prototipo devolvió en archivos CSV por cada modelo (categoría y nivel) la etiqueta de la categoría, la cantidad de muestras y la Accuracy alcanzada con el conjunto de prueba, luego del entrenamiento (Tabla 3).

Tabla 3. Precisión de los modelos entrenados para las 8 categorías de nivel 1.

Categoría de Nivel 1	Cantidad de muestras	Precisión
2	**66988**	**90.01**
3	**123743**	**81.98**
4	**190422**	**86.25**
5	**122752**	**81.82**
6	**6050**	**93.25**
7	**63249**	**91.52**
8	**50415**	**91.27**
9	**11**	**40.00**

Si bien la estrategia utilizada permitió obtener resultados razonables teniendo en cuenta los problemas antes citados con respecto al dataset y el tipo de clasificación, cabe aclarar que esta estrategia también favorece la expansión de errores de predicción para el caso de un falso positivo en Nivel 1 que causará la clasificación de ese sumario en un conjunto de categorías de nivel 2 que no corresponde.

6. Conclusiones y Trabajos Futuros

La gran cantidad de documentos legales que hoy en día se encuentran almacenadas electrónicamente son, en su mayoría, heterogéneos y de gran tamaño. Esto ha llevado al desarrollo de metodologías para procesamiento de dichos documentos y para la extracción de información útil con el objetivo de mejorar la recuperación de información relevante.

En este trabajo se presentó una estrategia de clasificación multinivel de documentos legales (fallos judiciales) que intenta hacer frente a las particularidades de la empresa editorial por un lado y de la tarea en sí misma por otro. A las dificultades propias de la clasificación de documentos se suma la necesidad de utilizar una estructura de clasificación con distintos niveles para los cuales a veces los ejemplos son insuficientes o directamente inexistentes. En la propuesta realizada se buscó sortear estas dificultades balanceando los conjuntos

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

de datos y utilizando una estrategia de múltiples regresores logísticos, con resultados aceptables que pueden ser mejorados reentrenando los regresores en caso de obtener más muestras en alguna categoría gracias a la división del problema que permite entonces reentrenar solo algunos regresores sin afectar al resto.

Los resultados obtenidos, en relación con la utilización que la empresa realizará de los mismos, son prometedores.

Como trabajo futuro se plantea la extensión de la estrategia para el nivel 3 del tesauro y estudiar la posibilidad de la generación automática de los sumarios.

References

1. Junejo, K. N., Karim, A., Hassan, M. T., & Jeon, M. (2016). Terms-based discriminative information space for robust text classification. Information Sciences, 372, 518-538.
2. Chen, X., Vorvoreanu, M., & Madhavan, K. (2014). Mining social media da-ta for understanding students' learning experiences. IEEE Transactions on learning technologies, 7(3), 246-259.
3. Liu, S. M., & Chen, J. H. (2015). A multi-label classification based approach for sentiment classification. Expert Systems with Applications, 42(3), 1083-1093.
4. Lita, L. V., Yu, S., Niculescu, S., & Bi, J. (2008). Large scale diagnostic code classification for medical patient records. In Proceedings of the Third Interna-tional Joint Conference on Natural Language Processing: Volume-II.
5. Yen, I. E. H., Huang, X., Ravikumar, P., Zhong, K., & Dhillon, I. (2016, June). Pd-sparse: A primal and dual sparse approach to extreme multiclass and multilabel classification. In International conference on machine learning (pp. 3069-3077). PMLR.
6. Jain, H., Balasubramanian, V., Chunduri, B., & Varma, M. (2019, January). Slice: Scalable linear extreme classifiers trained on 100 million labels for re-lated searches. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (pp. 528-536).
7. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.
8. Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward net-works are universal approximators. Neural networks, 2(5), 359-366.
9. You, R., Zhang, Z., Wang, Z., Dai, S., Mamitsuka, H., & Zhu, S. (2019). Atten-tionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. Advances in Neural Information Pro-cessing Systems, 32.
10. Ye, H., Chen, Z., Wang, D. H., & Davison, B. (2020, November). Pretrained generalized autoregressive model with adaptive probabilistic label clusters for extreme multi-label text classification. In International Conference on Machine Learning (pp. 10809-10819). PMLR.
11. Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., & Dhillon, I. S. (2020, August). Taming pretrained transformers for extreme multi-label text classification. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 3163-3171).
12. Vega-Marquez, B., Nepomuceno-Chamorro, I. A., Rubio-Escudero, C., & Ri-quelme, J. C. (2021). OCEAn: Ordinal classification with an ensemble ap-proach. Information Sciences, 580, 221-242.
13. Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning--based text classification: a comprehensive re-view. ACM Computing Surveys (CSUR), 54(3), 1-40.
14. Zubiaga, A. (2012). Enhancing navigation on wikipedia with social tags. arXiv preprint arXiv:1202.5469.
15. Vacek, T., Song, D., Molina-Salgado, H., Teo, R., Cowling, C., & Schilder, F. (2019, June). Litigation Analytics: Extracting and querying motions and orders from US federal courts. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstra-tions) (pp. 116-121).
16. Medvedeva, M., Vols, M., & Wieling, M. (2020). Using machine learning to predict decisions of the European Court of Human Rights. Artificial Intelli-gence and Law, 28(2), 237-266.
17. Xiao, C., Zhong, H., Guo, Z., Tu, C., Liu, Z., Sun, M., ... & Xu, J. (2018). Cail2018: A large-scale legal dataset for judgment prediction. arXiv preprint arXiv:1807.02478.
18. Chalkidis, I., Fergadiotis, M., Malakasiotis, P., & Androutsopoulos, I. (2019). Large-scale multi-label text classification on EU legislation. arXiv preprint arXiv:1906.02192.
19. Palau, R. M., & Moens, M. F. (2009, June). Argumentation mining: the detec-tion, classification and structure of arguments in text. In Proceedings of the 12th international conference on artificial intelligence and law (pp. 98-107).
20. Boella, G., Di Caro, L., & Humphreys, L. (2011). Using classification to sup-port legal knowledge engineers in the Eunomos legal document management system. In Fifth international workshop on Juris-informatics (JURISIN).
21. Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., & Lampos, V. (2016). Pre-dicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. PeerJ Computer Science, 2, e93.
22. Sulea, O. M., Zampieri, M., Malmasi, S., Vela, M., Dinu, L. P., & Van Ge-nabith, J. (2017). Exploring the use of text classification in the legal do-main. arXiv preprint arXiv:1710.09306.
23. Sulea, O. M., Zampieri, M., Vela, M., & Van Genabith, J. (2017). Predicting the law area and decisions of french supreme court cases. arXiv preprint arXiv:1708.01681.
24. Ji, D., Tao, P., Fei, H., & Ren, Y. (2020). An end-to-end joint model for evidence information extraction from court record document. *Information Processing & Management*, *57*(6), 102305.
25. Ji, D., Gao, J., Fei, H., Teng, C., & Ren, Y. (2020). A deep neural network model for speakers coreference resolution in legal texts. Information Pro-cessing & Management, 57(6), 102365.
26. De Araujo, P. H. L., de Campos, T. E., Braz, F. A., & da Silva, N. C. (2020, May). VICTOR: a dataset for Brazilian legal documents classification. In Proceedings of the 12th Language Resources and Evaluation Confer-ence (pp. 1449-1458).
27. Domingues, L. E. R., Ponciano, J. R., Nonato, L. G., & Poco, J. (2022). Le-galVis: Exploring and Inferring Precedent Citations in Legal Documents. IEEE Transactions on Visualization and Computer Graphics.
28. Li, Z. (2019, January). A classification retrieval approach for English legal texts. In 2019 International Conference on

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

Intelligent Transportation, Big Data & Smart City (ICITBS) (pp. 220-223). IEEE.
29. Song, D., Vold, A., Madan, K., & Schilder, F. (2022). Multi-label legal docu-ment classification: A deep learning-based approach with label-attention and domain-specific pre-training. Information Systems, 106, 101718.
30. Chen, H., Wu, L., Chen, J., Lu, W., & Ding, J. (2022). A comparative study of automated legal text classification using random forests and deep learn-ing. Information Processing & Management, 59(2), 102798.
31. Priyadarshini, R. (2021). LeDoCl: A Semantic Model for Legal Documents Classification using Ensemble Methods. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(9), 1899-1908.
32. Perezzini, L., Casali, A., & Deco, C. (2020). Sistema de soporte para la recu-peración de normativas en la ingeniería legal. In XX Simposio Argentino de Informática y Derecho (SID 2020)-JAIIO 49 (Modalidad virtual).
33. Khyani, Divya & B S, Siddhartha. (2021). An Interpretation of Lemmatization and Stemming in Natural Language Processing. Shanghai Ligong Daxue Xuebao/Journal of University of Shanghai for Science and Technology. 22. 350-357.
34. Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: a statistical framework. International journal of machine learning and cybernetics, 1(1), 43-52.
35. Bafna, P., Pramod, D., & Vaidya, A. (2016, March). Document clustering: TF-IDF approach. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 61-66). IEEE.
36. Kramer, O. (2013). K-Nearest Neighbors. In: Dimensionality Reduction with Unsupervised Nearest Neighbors. Intelligent Systems Reference Library, vol 51. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-38652-7_2
37. (2011). Logistic Regression. In: Sammut, C., Webb, G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8_493
38. https://scikit-learn.org/stable/ ultimo acceso Junio 2022.