DOI: 10.5748/18CONTECSI/PSE/DSC/6815

LEGAL DOCUMENT CLUSTERING WITH TF-IDF VECTORS, KOHONEN MAP AND K-MEANS

João Pedro Da Silva Lima - UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE - Orcid: https://orcid.org/0000-0002-5706-6065

José Alfredo Ferreira Costa - UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE - Orcid: https://orcid.org/0000-0002-1290-6454

Diógenes Carlos Albuquerque Araújo - UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE - Orcid: https://orcid.org/0000-0002-9859-9255

Addresses the usage of clustering algorithms applied to judicial documents in brazilian Portuguese containing process movements. The paper uses a dataset in brazilian Portuguese. The task is explored using TF-IDF vectors from a database of 30,000 process movements provided by the Tribunal de Justiça do Rio Grande do Norte (TJRN). The proposed model was capable to develop consistent and meaningful clusters, and the new metric was helpful to evaluate and to interpret these clusters. The paper also approaches new Natural Language Processing (NLP) specific measure to evaluate inter-cluster similarity, based on the most important words of each cluster. The number of cases in Brazilian Judiciary System reached the mark of 77 million in 2019, most of these lawsuits are being processed in a digital way These facts create the ideal scenario for the implementation of machine learning models, which can automate various types of activities and increase celerity.

Keywords: Artificial Intelligence, Unsupervised learning, Clustering, Natural Language Processing, Judicial

LEGAL DOCUMENT CLUSTERING WITH TF-IDF VECTORS, KOHONEN MAP AND K-MEANS

1. INTRODUCTION

The digital revolution has accelerated the increase in the data volume and complexity in many areas such as industry, finance, natural sciences, linguistics, and social sciences. The judiciary system in many countries, such as Brazil, is a challenge in terms of storage and processing data, as the number of cases was approximately 77 million in 2019, as stated in a Brazilian National Justice Council (CNJ) (CNJ - Conselho Nacional de Justiça, 2020). Due to the strong judicial digitalization policy, most of these cases are processed entirely in a digital way, e. g., on the PJe platform, which is used by many Brazilian courts. However, process celerity and efficiency are needed. This scenario is suitable for machine learning techniques, given that the big data could be used for training models that could aid many activities, such as automatic document classification and similarity document organization. The general idea is to automate some steps in the process, such as repetitive tasks, aiming the judicial effectiveness and celerity. Brazil already counts on various projects targeting the use of Artificial Intelligences (AI) in justice, with applications focusing on automation and better information retrieval. Projects like SINAPSES (CNJ - Conselho Nacional de Justiça, 2019) platform could lead to robust and easy to use artificial intelligence models. SINAPSES were initially developed by the Tribunal de Justiça de Rondônia (TJRO) in collaboration with CNJ with goals including to create a collaborative environment, which can condense and share machine learning models developed in several Brazilian courts.

Despite other scientific data mining and AI usage, text or document processing in legal systems engage major challenges, e. g., unstructured and unlabeled data are very common, with a diversity of text and situations. Not only the amount of information but its complexity, which can be very difficult to extract, since information may be distributed in different files with different formats, usually not algorithm friendly. Also, due to its large volume, manual processing analysis is unfeasible. In this context, the study of unsupervised algorithms becomes essential, as they can learn patterns and obtain representations of the bulk data, transforming them into more dense information for both humans and machines.

Clustering is the process of discovering groups within the data, based on similarities, with a minimal, if any, knowledge of their structure. The collection of data $X=\{x_1, x_2, \ldots, x_n\}$ can be represented as a set of points in a multidimensional vector space, $\mathbf{X} \subset \mathbf{R}^p$. The objective of clustering is to find a convenient and valid organization of the data, based on the inherent structure and relationships among the patterns. It may be seen as the process of dividing the p-dimensional pattern space in a way that those patterns in a given cluster are more similar to each other than the rest (Costa and de Andrade Netto, 2001; Jain et al., 1999).

The detection of natural groups within a data set, the clustering, is a task that has great utility in the areas of data mining and information retrieval. In the juridical environment, the ability to group similar texts is especially important, as it can bring great efficiency to various tasks with similar cases leading to equity and social justice (Lu et al., 2011; Raghav et al., 2015 b; Mandal et al. 2021; Francesconi and Passerini, 2007). Furthermore, cluster labels can be used as information for other machine learning models.

The self-organizing feature map (SOM) is the most used neural network inthe unsupervised learning paradigm (Kohonen, 1990). It has been widely studied as a software

tool for visualization of high dimensional data. Important features include information compression while preserving topological and metric relationship of the primary data items. Visualization of neuron's relations can be done using the Unified distance matrix (U-matrix), which is a neuron-distance image (Ultsch, 1993).

Many machine learning techniques, especially in the area of Deep Learning, have proven to be very efficient for numerically capturing the syntax and semantics of large and complex texts, such as BERT and ULM FIT for large texts and Word2Vec (Mikolov et al., 2013) for unique words. However, much of the efficiency comes at the cost of a complex model to be trained and practically impossible to be understood.

In the legal environment, it is extremely important that the algorithms can be constantly retrained, to avoid wrong decisions based on old data, and audited, so that you can give a support the justification of your decision.

This paper proposes the application of a twostep clustering and data visualization approach, by using SOM and K-means algorithms to group TF-IDF vectors generated from a corpus of documents. It is also proposed a new NP specific measure to evaluate inter-cluster similarity, based on the most important words of each cluster.

In particular, unsupervised learning tasks are especially difficult to train and evaluate, since we do not have a "target" to which we can compare. However, it is still possible to define a metric that assesses the general behavior that the grouped data should have. In our experiments, we propose the use of quantitative and qualitative metrics specific to the NLP scope, in an attempt to make a more adequate analysis and always valuing interpretability.

Our experiments were conducted on a dataset made available by the Tribunal de Justiça do Rio Grande do Norte with 30, 000 judicial processes. The base is divided between 10 classes, which help us to evaluate the clustering in addition to the unsupervised metrics and to verify if the labels created by the algorithm are coherent with the originals, given by humans.

2. RELATED WORK

The goal of grouping similar texts implies many important judicial tasks. For Example, through the incident of resolution of repetitive demands (CPC, 2015), the same actions will be frozen in the first instance until the Court of Justice or the Regional Federal Court decides the issue and has the same decision applied in all cases.

(Kim et al., 2020) used neural networks-based techniques to simultaneously transform and create soft clusters of texts. (Lu et al., 2011) demonstrated a method for large scale soft clustering with topic segmentation in documents of legal domain, successfully managing to create an effective algorithm for retrieving and processing information.

Still in the legal environment, (Wagh, 2014) demonstrates a grouping technique based on a K-Means variant for a document search system. (Rabelo et al., 2019), utilize cluster and classification in conjunction with advanced Deep Learning Techniques for case law entailment, statute law retrieval and entailment.

(Raghav et al., 2015a) described a method to perform text and citations clustering from legal judgments. The authors explore the idea that citation information in legal judgments could be utilized for efficient search and to establish similarity between judgments. The authors reported that the proposed clustering approach was able to find similar judgments by exploiting citations in paragraph links.

Recently, (Mandal et al., 2021) presented unsupervised approaches for measuring textual similarity between legal court case reports. The authors address measuring document similarity as the most crucial aspects of any document retrieval system which decides the speed, scalability and accuracy of the system. They investigated the performance of 56 different methodologies for computing textual similarity across court case statements when applied on a dataset of Indian Supreme Court Cases, including BERT (Devlin et al., 2018) and Law2Vec (Chalkidis and Kampas, 2019). The authors reported that traditional methods (such as the TF-IDF) performed better than the more advanced context-aware methods

3. THEORETICAL BASIS

In this section, we explain the machine learning algorithms used to build our model. We also introduce the metric proposed to evaluate cluster similarity in NLP tasks.

3.1 TF-IDF

TF-IDF is a classic textual vectorization algorithm, which is based on two coefficients: text frequency (TF), and inverse documental frequency (IDF), to build numerical vectors. The algorithm assigns an weight ji to each word of the document d according to the equation below:

$$p_{ji} = \mathrm{TF}(w_i, d_j) \log(\mathrm{IDF}(w_i))$$

Where $\mathrm{TF}(w_i, d_j)$ is the frequency of the word *i* in the text *j*, and IDF(wi) the inverse of the documental frequency of the word throughout the corpus . The representation created by this algorithm will be a vector of size $|V|$, where V represents the corpus vocabulary. The idea is that the weights can translate the importance of each word to the given document (Ramos et al., 2003), based on its rarity.

3.2 Kohonen's Map

Kohonen's map (Kohonen, 1990) is a type of Self-Organizing Map that can represent a p-dimensional space to a set of neurons. It consists of a two-dimensional grid of neurons capable of adapting to a data set. Neurons are randomly initialized in the original space and are interactively optimized based on a metric for distance, usually euclidean, between them and each data entry.

The SOM training is accomplished by comparing a data set entry with each neuron. We define the Best Match Unit (BMU) as the neuron *n* of index *c*, such that:

$$||x - n_c|| = \min_i \{||x - n_i||\}$$

Then, the BMU and its neighbors are moved closer to the data entry. The rule for the movement is:

$$n_i^{t+1} = n_i^t + h_{c,i}^t \cdot [x - n_i^t]$$

Where t denotes the time step of training and x denotes an entry drawn from the data set. $h_{c,i}^{t}$ is the neighborhood kernel around the BMU *c* at time t, a non-increasing function of the grid distance between *i* and *c*, and time. The idea is to gradually bring the neurons closer to the large dense areas in the original space. Finally, each entry in the dataset will be represented by its BMU. This 'micro-clustering' made by the Kohonen map allows us to describe the distribution of the original data in a much more efficient way, given that we summarize a set of closed points to a single point in space. This last characteristic is only true when we have a total of neurons less than the total entries in the data set, which is our case for all experiments made.

3.3 K-Means

K-Means (Jain et al., 1999) is one of the most famous clustering algorithms, its purpose is to partition data into K distinct groups. It aims to minimize the sum of mean squared distances between each point in the data set to its closest centroid. The most common way to find the best centroids is to iteratively move the centroids in the Expectation Maximization approach.

3.4 U-Matrix

The U-matrix method enables visualization of the topological relations of the neurons in an organized SOM (Ultsch, 1993). The basic idea is to use the same metric that was used during the learning to compute distances between adjacent reference vectors. These distances can be visualized in a 3D-plot (or in an image), in which valleys correspond to map units that are similar. High values in the U-matrix encode dissimilarities between neurons and correspond to cluster borders.

3.5 Coincidence window of important words

For us, it is not enough that each group satisfies the notion of Euclidean distance previously established, it is also necessary that we have some interpretable sense in the grouped documents, that is, that we can identify a pattern that justifies the decision of the algorithm. This is important not only to assess their effectiveness, but also to ensure greater transparency, avoiding black-boxes algorithms.

In a real application, the volume of documents is probably very large, which makes it impossible to manually analyze each text. One solution found to evaluate the clusters was, in each of them, extract and visualize the top-n terms with the highest mean TF-IDF. This set of terms must contain the general rule of that cluster, and it is up to the context to decide whether or not there is sense behind them. As it is a text-based evaluation, the visualization is simple to understand and does not require a great theoretical basis.

Another characteristic of this method is that we can evaluate the similarity between two different clusters by comparing the respective top-n's.

Taking this into account, we propose a similarity score based on a term coincidence window: First, we consider the top-n most important words for

Taking this into account, we propose a similarity score based on a term coincidence window:

First, we consider the top-n most important words for clusters c_i and c_j. In our experiments, we consider the importance of a word in a cluster as the average of its

TF-IDF coefficients over all documents of that cluster. Then, for each word in the intersection of these sets, we retrieve its position in each top-n. If these two positions are close, i. e., if they differ, in maximum, by $window\_size$, we add +1 point to clusters c_i and c_j similarity. Figure 1 shows an example of this process for $n = 6$ and $window\_size$ = 1.

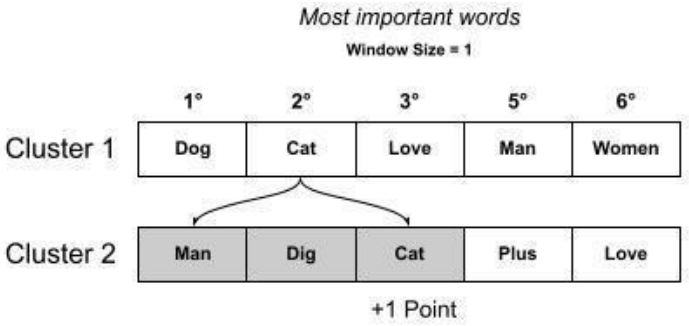

Figure 1: Example of word coincidence with window size = 1
Source: Author

4. TASK DESCRIPTION

For our model, it was decided to create a hierarchical clusterer, stacking two different clustering algorithms, Kohonen Map and KMeans. The Kohonen Map Is a self-organizing map capable of reducing the dimensionality of the data and performing clustering simultaneously. The main advantage of this algorithm is that we are able to preserve the original topology of the data while performing a pre-cluster and reducing the total number of entries.

Each neuron on the Kohonen map can be understood as an individual cluster. However, there is usually a very large number of neurons, which creates very small groups with no practical meaning. Therefore, a post-processing task is performed with K-Means, grouping the neurons on the map. Then, each entry is assigned with the same label as its BMU. The result is larger and more interpretable groups.

Our proposal was to create a transparent, updatable and auditable algorithm. Each step in the pipeline helps to ensure these characteristics. The TF-IDF is a classic textual vectorization process based on syntax and its main characteristic is the ability to assign a weight of importance to a term based only on its frequency in texts and in the corpus.

Both Kohonen's Map and K-Means follow similar, easy-to-understand assumptions, which can be summarized in iteratively moving 'clusters' to densely populated areas of the data set. In addition, these two algorithms have incremental learning, that is, if the database is updated, they can adapt to the new information, without the need to train completely from scratch.

5. DATASET

The dataset used is a corpus with 30, 000 documents of sentences to procedural movement at the Tribunal de Justiça do Rio Grande do Norte. Each entry is accompanied by a label,

out of a total of ten, which represents which class of movement it belongs to. These labels will be used to evaluate the model's ability to group texts that we already consider part of the same group. All classes contain 3, 000 documents. Table 1 provides a description of the labels.

Label	Name(pt)	Name(en)
196	Extinção da execução ou Cumprimento da sentença	Discharge of the execution or Judgement enforcement
198	Acolhimento de embargos de declaração	Acceptance of clarification motions
200	Não Acolhimento de embargos de declaração	Not Acceptance of clarification motions
219	Procedência	Validity
220	Improcedência	Denial
339	Liminar	Preliminary
458	Abandono de causa	Abandonment of the claim
461	Ausência das condições da ação	Absence of the action's conditions
463	Desistência	Discontinuance
785	Antecipação de tutela	Interlocutory remedy

Table 1: Dataset classes description

Before being fed to the model, the dataset has passed by a cleaning step, described below:

– The text has been converted to lowercase
– Were removed special symbols, punctuation and numbers
– The Portuguese stopwords have also been removed
– The words present in less than 2. 1% of the documents and in more than 90%of the documents have been removed. These values were obtained empirically.
TF-IDF vectors are especially sensitive to vocabulary size and may not scale well for very large datasets. The percentage cuts helped to dramatically reduce the dimensionality of these vectors, which speeds up the entire pipeline process. In the end, we are left with a vocabulary of 3, 359 terms.

6. RESULTS

As previously mentioned, the pre-processing of the data left us with a vocabulary of 3, 359 words, that is, our TF-IDF vectors have dimension $|V| = 3,359$.
The Kohonen map used has 2, 500 neurons distributed in a 50x50 rectangular grid and is initialized randomly in the original space. An advantage of the map is that it can give us a hint of how many natural clusters exist and the distribution of neurons through the U-Matrix (Ultsch, 2003) plot. Figure 2 shows the U-matrix for the 50x50 trained SOM after 50 epochs. Lighter pixels correspond to higher distances in the U-matrix, related to cluster borders

As also mentioned previously, K-Means is responsible for grouping the neurons in the map, bringing the number of clusters to a more interpretable value. The test described here was performed with a number of clusters equal to 8. This number of clusters was found after a set of simulations. Figure 3 shows the best grid partition found after 250 K-means runs. As we can see, the clusters match the borders delimited in U-matrix plot.

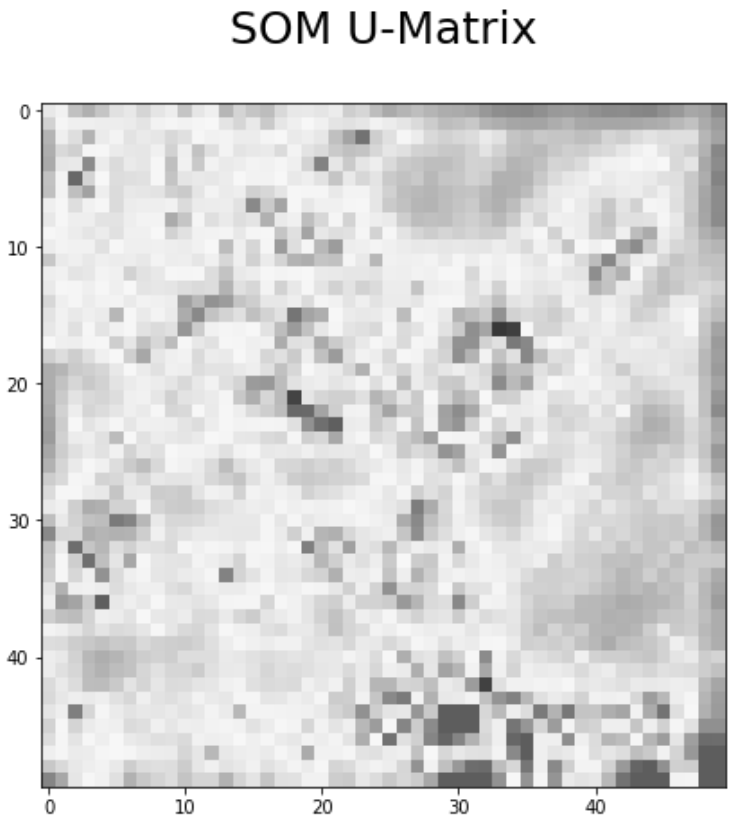

Figure 2: SOM U-Matrix of the data
Source: Author

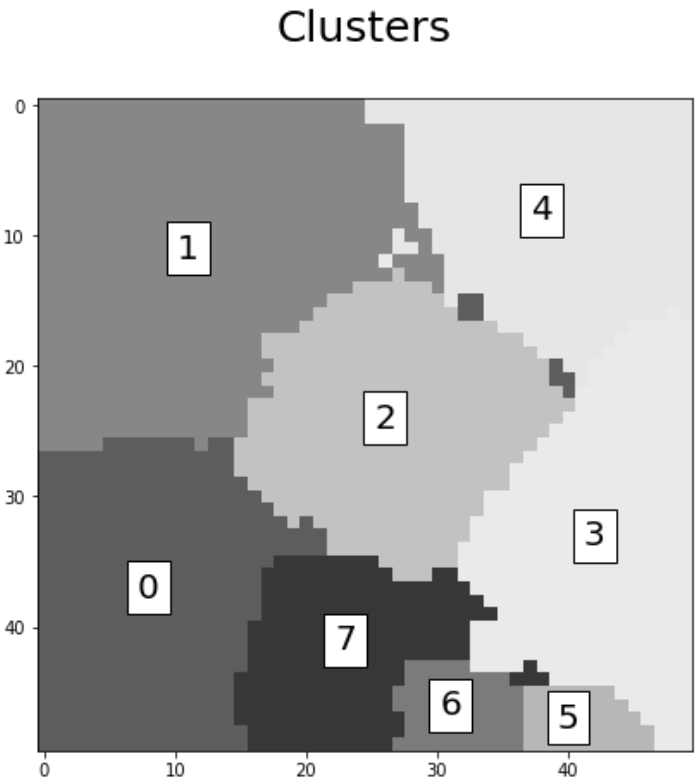

Figure 3: Clusters over SOM U-Matrix
Source: Author

As described in 3. 5, we apply specific methods from the NLP area to evaluate the results. At first, we visualize the most important terms in each of them. This method can give us a clue about the nature of a cluster's data. This approach has a strong qualitative character and requires prior knowledge of the general content of the database.

It is extremely important that we can logically evaluate the clusters because, especially in the case of models made to the legal environment, transparency and auditing capacity are important factors that must be considered.

Table 2 shows that most clusters are following some detectable direction. Some conclusions can be drawn:

Cluster 0	Cluster 1	Cluster 2	Cluster 3
danos	desistência	execução	fiscal
morais	natal	exequente	execução
consumidor	art	sentença	embargos
dano	resolução mérito	executado	execução fiscal
art	autor	cumprimento	fazenda
danos morais	mérito	natal	sentença
autora	resolução	obrigação	mossoró
indenização	feito	devedor	art
valor	extinto	extinção	pública
autor	natal rn	cumprimento senteça	interesse
Cluster 4	**Cluster 5**	**Cluster 6**	**Cluster 7**
tutela	embargos	embargos	embargos
liminar	embargos declaração	embargos declaração	sentença
medida	declaração	declaração	natal
natal	omissão	sentença	embargos declaração
autora	contradição	omissão	declaração
antecipação	obscuridade	embargante	embargante
pedido	sentença	contradição	omissão
parte autora	embargante	interpostos	valor
decisão	decisão	decisão	natal rn
apreensão	erro	natal	decisão

Table 2: Top-10 most important words for each cluster in Portuguese

– In cluster 0, we can see terms such as ‘indenização’, ’danos morais’, ’danos’ and ’valor’, which are strongly related to moral damage actions.
– In cluster 1, the words ’resolução’, ’desistência’ and ’extinção’ are related to movements involving the withdrawal or extinction of lawsuits.
– In cluster 2, the words ’extinção’, ’cumprimento’ and ’execução’ refer to the movements to terminate the execution or enforcement of the sentence.
– Cluster 3 probably refers to actions related to money, due to the presence of ’fazenda’, ’embargos’ and ’fiscal’.
– Cluster 4 refers to injunction and guardianship actions, noted by the presence of the words ’liminar’, ’pedido’ and ’tutela’.
– Clusters 5, 6 and 7 have very similar words, and probably refer to movements with declaration of embargoes.

Another important analysis to be done is the similarity between the clusters. Although we can manually observe the 10 most important words of each one and draw conclusions, this task can become expensive and time consuming for large groups, so it is interesting that we can quantize and automate this process. For This, we apply the metric discussed in section 3. 5 with a top-500 list of each cluster and a window of size 5.

As we can see from Figure 4, most clusters have between 4 and 9 coincident terms, while clusters 5, 6 and 7 have 16, which is almost double. This helps us to confirm our suspicion raised in table 2, that the content spread among them is probably the same, and perhaps it should belong to a single cluster. This proximity can also be seen back in the U-matrix, where we can note that these clusters are small and close to each other.

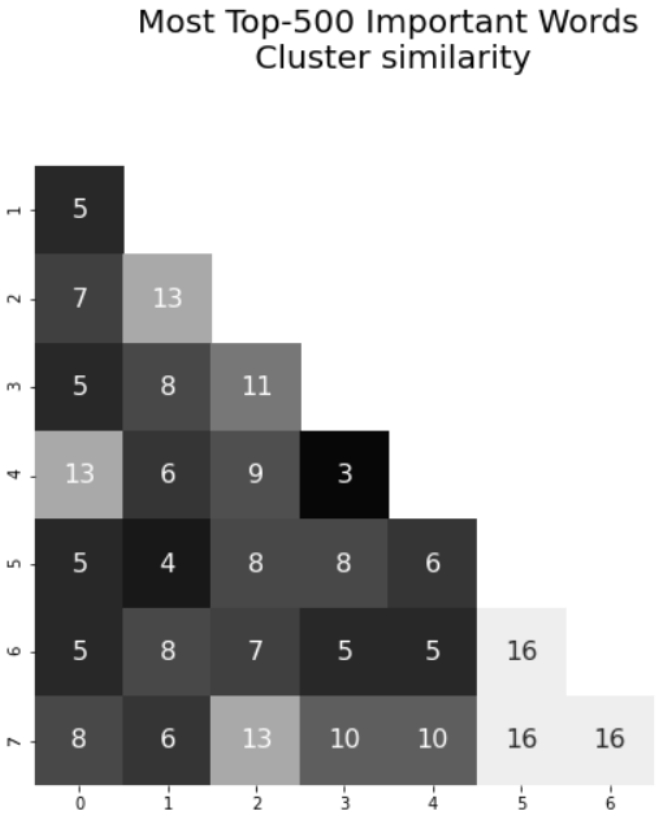

Figure 4: Number of word coincidences for each cluster pair
Source: Author

Up to this point, we have focused entirely on evaluating our results in a completely unsupervised manner. The considered criteria were able to deliver consistent information about the original data, making the clusters transparent and easy to interpret.

As mentioned in section 5, the data used already have their own labels, i. e., original classes assigned by humans. Our task belongs to the area of unsupervised learning, so we are not going to get too deep into supervised metrics. However, In Figure 5 we visualize the contingency matrix generated with the results of the model versus the original classes.

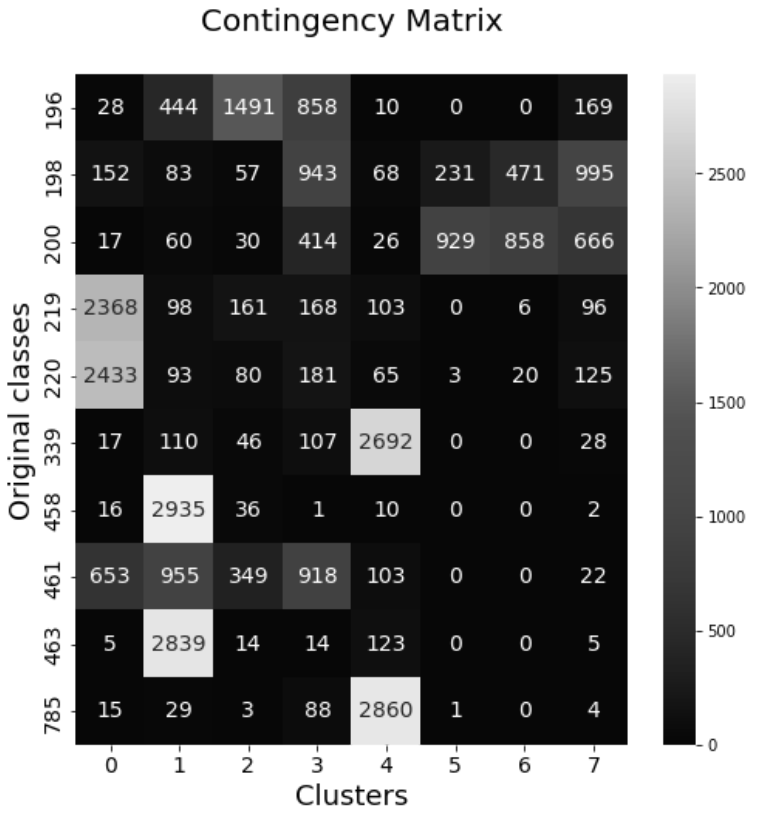

Figure 5: Contingency matrix for original classes *versus* cluster labels
Source: Author

As we can see, the classes are distributed relatively well among each of the created clusters. An event that we can notice is the frequent mixture of two distinct classes in one or more clusters. This apparent "confusion" of the model can be explained by the strong similarity between the procedural movements that suffer from this anomaly.

Movement pairs 219 and 220, 198 and 200, are documents that deal with exactly the same type of subject and differ only in the final decision of the sentence. That is, the members of these pairs differ only by a few terms that indicate denial or acceptance, and the rest of their content is mostly identical. Pairs 339 and 785, 458 and 463 are similar judicial movements, with similar outcomes or effects, or triggered in similar cases.

7. CONCLUSION

This article describes the creation and evaluation of a machine learning model capable of grouping similar judicial texts. The task of textual clustering is particularly challenging, as it mixes the difficulties of manipulating complex unstructured data with the difficulties of training and validating unsupervised models.

The legal feature of the texts brings an even greater difficulty, its specific vocabulary and unique formatting makes some generic vectorization models, such as Word2Vec and Doc2Vec, have difficulties in representing their Corpus. Although TF-IDF vectors are very simple, they have proved to be extremely effective in representing our database.

Our model manages to perform the task of clustering and generate good results, with representative clusters and endowed with interpretable meanings, in addition to meeting the imposed restrictions for effective application in the legal environment. The results demonstrated that it is possible to implement reliable, auditable, and transparent unsupervised machine learning model.

In addition to the model, we approached techniques for assessing the quality of clusters focused on the NLP scope. The new metric proposed, based on coincidence windows on the most important terms of the clusters, combines transparency, scalability and ease of interpretation.

8. FUTURE WORK

A possible approach to future work is the adoption of more interpretable hierarchical methods. Our implementation can be understood as hierarchical clustering, but the first layer of clusters, generated by Kohonen Map, is useful only for K-Means. It would be interesting to create a method capable of generating different levels of clusters and bringing different levels of abstraction to the same dataset. We also intend to apply our model with other text vectorization algorithms and topic modeling.

REFERENCES

CPC (2015). LEI N° 13.105 de 16 de março de 2015.
Available at http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13105.html

Chalkidis, I. and Kampas, D. (2019). Deep learning in law: early adaptation and legal word embeddings trained on large corpora. Artificial Intelligence and Law, 27(2):171–198.

CNJ - Conselho Nacional Justiça (2019). SINAPSES-Documentação em desenvolvimento.
Available at http://docs. pje. jus. br/servicos-auxiliares/sinapses/sinapses-manual. html.

CNJ - Conselho Nacional de Justiça (2020). Relatório Justiça em Números.
Available at https://www. cnj. jus. br/pesquisas-judiciárias/justica-em-numeros/.

Costa, J. A. F. and de Andrade Netto, M. L. (2001). Clustering of complex shaped data sets via Kohonen maps and mathematical morphology. In Data Mining and Knowledge Discovery: Theory, Tools, and Technology III, volume 4384, pages 16–27. International Society for Optics and Photonics.

Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810. 04805.

Francesconi, E. and Passerini, A. (2007). Automatic Classification of provisions in legislative texts. Artificial Intelligence and Law, 15(1):1–17.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. ACM computing surveys (CSUR), 31(3):264–323.

Kim, J., Yoon, J., Park, E., and Choi, S. (2020). Patent document clustering with deep embeddings. Scientometrics, pages 1–15.

Kohonen, T. (1990). The self-organizing map. Proceedings of the IEEE, 78(9):1464–1480.

Lu, Q., Conrad, J. G., Al-Kofahi, K., and Keenan, W. (2011). Legal Document clustering with built-in topic segmentation. In Proceedings of the 20thACM international conference on Information and knowledge management, page 383–392.

Mandal, A., Ghosh, K., Ghosh, S., and Mandal, S. (2021). Unsupervised approaches for measuring textual similarity between legal court case reports. Artificial Intelligence and Law, pages 1–35.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. arXiv preprint arXiv:1310. 4546.

Rabelo, J., Kim, M. -Y., and Goebel, R. (2019). Combining similarity and transformer methods for case law entailment. In Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law, pages 290–296.

Raghav, K., Balakrishna Reddy, P., Balakista Reddy, V., and Krishna Reddy, P. (2015a). Text and citations based cluster analysis of legal judgments. In Prasath, R., Vuppala, A. K., and Kathirvalavakumar, T., editors, Mining Intelligence and Knowledge Exploration, pages 449–459, Cham. Springer International Publishing.