Topic-Modelling Based Approach for Clustering Legal Documents

Aayush Halgekar[1[0000-0002-3515-5665]], Ashish Rao[1[0000-0002-6676-8819]], Dhruvi Khankhoje[1[0000-0003-3238-7687]], Ishaan Khetan[1[0000-0002-4267-3938]] and Kiran Bhowmick[1]

[1] Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

{aayush.halgekar, ashish.rao, dhruvi.khankhoje, ishaan.khetan, kiran.bhowmick}@svkmmumbai.onmicrosoft.com

Abstract. The justice system has been institutionalized around the world for a long time, increasing the number of resources available for and in this field. The colossal increase in dependency on the World Wide Web is commensurate to the increase of digitization of documents. Along with this growth has come the need for accelerated knowledge management—automated aid in organizing, analysing, retrieving and presenting content in a useful and distributed manner. For a fair, cogent and strong legal case to be built, the individual fighting the case must have access to case documents from not only several years ago but also a few months ago. Any particular part of any of these various cases that received a verdict in the previous years could be beneficial to the individual's case. Considering all these factors, it is evident to develop a search engine for legal documents which will provide the user with all the relevant documents it requires. Moreover, unlike widely accessible documents on the Internet, where search and categorization services are generally free, the legal profession is still largely a fee-for-service field that makes the quality (e.g., in terms of both recall and precision) a key differentiator of provided services. This paper proposes a unique approach to group these documents by clustering them using the Mini Batch K-Means algorithm on dimensionally reduced sentence embeddings generated with the use of DistilBERT and UMAP. The proposed approach has been compared to state of the art topic modelling and clustering approaches and has outperformed them.

Keywords: Document Clustering, Sentence Embeddings, Mini Batch K-Means, Topic Modeling, UMAP.

1 Introduction

The internet contains a large amount of legal documents and while it facilitates the storage of documents and other data digitally, protecting them from physical damage and loss of information, it has its unique set of problems. One such problem being finding related articles or proceedings. In addition to the previous statement, the presence

of myriad documents and resources has engendered a need for a search engine to enable discovery of the required documents in an optimized manner.

The legal domain is one such domain where the legal proceedings and articles have been digitized due to the increasing number of documents produced every day. In the legal sphere, simultaneously multiple cases are fought and could be setting a precedent for one another. There are innumerable cases that have already been shut which still might play a small but extremely relevant role in an ongoing case. Additionally, there is also a possibility of a past case having an immense impact, but due to the disorganization and unshattered availability of these documents, they get missed. These problems indicate the urgent need for a way to group documents related to a topic and organize them better in order to be stored and retrieved efficiently. Considering all these characteristics and circumstances, a clustering of legal documents will enable organized, well-timed and quick search of these documents. Clustering is an unsupervised machine learning approach where one draws inferences from datasets consisting of input data without labelled responses. Generally, it is used as a process to find meaningful structure, explanatory underlying processes, generative features, and groupings inherent in a set of examples. By grouping related pages into one category, clustering is used to assist users in navigating, summarizing, organizing, and improving the search results of the huge amounts of textual data [1].

In the context of this paper, a clustering algorithm will serve to group topically similar legal documents together. This will not only help the legal community for knowledge-management but also any individual who fails to understand the jargons of the legal community in obtaining the required information by browsing through similar articles and not having to explicitly search for them. To allow the legal community to retain both their broad coverage and productivity in their practice of law it is essential to adequately cluster legal documents and facilitate their organized access.

The proposed approach aims to create clusters that are internally cohesive but are extremely different from other clusters. The methodology of clustering text documents is to use their term frequency value to group the documents having akin terms in one cluster. The terms act as a measure of topic or the contents of a document and accordingly assign them to a cluster. Furthermore, with the recent advancements in the field of Natural Language Processing, context-aware approaches which not only take into consideration the mere words present in a document but also their relationship with other words in a sentence have also been explored in this paper.

The key contribution of this work involves successfully developing a topic-modelling based clustering approach for the grouping of legal documents to create pure and cohesive clusters that outperform existing topic modelling and clustering models.

The paper discusses previous and related work in Section 2, the proposed methodology and steps employed during implementation of the proposed system in Section 3. Section 4 explains the dataset used, its analysis and comparative modelling and Section 5 discusses the results of various existing models and the proposed model.

2 Related Work

Document Clustering algorithms have traditionally relied on algorithms used to perform unsupervised clustering on numerical or tabular data. The main premise behind such algorithms is the use of vectorization of the documents present and applying a traditional clustering algorithm. In the case of [2], this approach is demonstrated along with other approaches such as a heuristic variant and a fuzzy variant of the traditional K-Means. In [3], authors introduce an improvement to the traditional K-Means algorithm which helps to provide a faster and more efficient search algorithm which provides a more global optimal solution. Similarly, [4] employs a Particle Swarm Optimization (PSO). [5] works on improving the traditional K-Means algorithm by dimensionality reduction which helps to make it faster and efficient. On the other hand, [6] focuses on the Agglomerative clustering algorithm which is another traditional clustering algorithm employed usually for numerical data.

Authors in [7], [8], [9] focus on graph-based solutions to solve the problem of logical relevance during document clustering. In [7], the paper uses a taxonomy-based approach on legal documents. [8] and [9] employ a graph-based solution based on background knowledge in the form of ontologies and along with it, K-means clustering.

Some other approaches which try to subsume a context-aware component in them are based on the usage of different types of similarity metrics. The idea behind these approaches is to check for the similarity of a document with the other documents present in the corpus. Traditionally, similarity metrics such as the Euclidean distance or the Manhattan distance, have been used. Though these similarity metrics may be helpful in providing the similarity between numerical values, they fail to capture the similarity between similar words. In order to tack-le this problem, various methods have made the use of the cosine similarity of the vectors created on the basis of words in the document. However, in [10], the authors demonstrated that the use of cosine similarity alone is insufficient as it does not take into consideration the difference in magnitude of two vectors. In order to tackle this problem, they introduced a new similarity metric which provides a more accurate similarity prediction as compared to traditional similarity metrics. In [11], another similarity metric has been introduced known as the ordered weighted average (OWA) in order to provide a similarity value.

In their paper, Xie and Xing [12] talk about the possible integration of the concept of topic modelling with that of document clustering. Their paper proposes a new model based on the LDA model for topic modelling. Based on the topics generated by the model, the clustering of documents takes place. The LDA model is used specifically for the clustering of legal documents in [1] which clusters the documents based on the highest probability for a legal document to belong to a topic. In [1], a more complex variant of the traditional LDA algorithm for topic modelling is adopted, which is known as Hierarchical Latent Dirichlet Allocation. This variant extends the traditional LDA algorithm by adding the concept of hierarchy and predicting not only the topics subsumed in a document but also topic trees of the various documents.

In order to provide further advanced solutions to the defined problem, [13] works with sentence embeddings paired with a dimensionality reduction technique followed

by a density-based clustering approach. [14] proposes a modification on [13] by adding c-TF-IDF which helps to extract the main words and understand the topics of the generated clusters. This paper proposes a new methodology which performs better than existing architectures.

3 Methodology

This section describes the methodology implemented. Each section addresses a component in the overall procedure followed. The proposed methodology is illustrated in Fig 1.

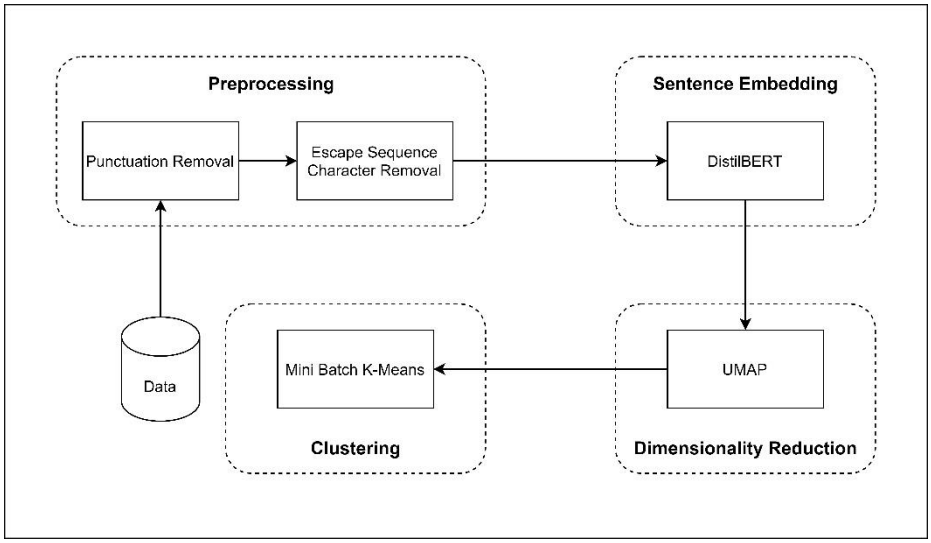

Fig. 1. Process Workflow

3.1 Preprocessing

For pre-processing, a pre-processing pipeline involving punctuation removal and escape sequence character removal is performed. Stopwords have not been removed in order to preserve the relationship of words. Additionally, lemmatization or stemming have been avoided in order to preserve the lexical structure of the data.

3.2 Sentence Embeddings

Pre-processed documents are then embedded into numeric values using a Sentence Transformer. The DistilBERT language model is used under the hood by the Sentence Transformer to generate the sentence embeddings. The sentence-transformers python module is used for the implementation [15].

3.3 Dimensionality Reduction

The embedded documents are spread across various dimensions which makes it more difficult to cluster. The Uniform Manifold Approximation and Projection or UMAP [16] is used to reduce the dimension of the embeddings to 5 dimensions. Since the document vectors in high dimensional spaces are very sparse, dimension reduction might help in finding dense areas. It is chosen for dimensionality reduction in this paper, as it preserves local and global structure, and is able to scale to very large datasets [13]. The most important hyper-parameter out of the several UMAP has to determine its way of performing dimension reduction is the number of nearest neighbours. This parameter is responsible for maintaining balance between preserving global structure versus local structure. More emphasis is put on global over local structure preservation by larger values.

3.4 Clustering

The dimensionally reduced embeddings are then clustered using the Mini Batch K-Means algorithm [17], which is a variant of the traditional K-Means algorithm. The algorithm employs the use of a series of iterations to continuously update the centroids of the clusters to get a dynamic cluster formation. In each new iteration, the document is assigned a cluster based on the current position of the centroids. Once the document is assigned to a cluster the position of the centroid is recalculated based on gradient descent which is faster than the traditional K-Means algorithm.

It uses mini-batches to reduce the computation time in large datasets. Additionally, it attempts to optimize the results of the clustering. To achieve this, the mini batch k-means takes mini-batches as an input, which are subsets of the whole dataset, randomly.

4 Experimentation

4.1 Dataset Description

The dataset used in the paper is an open-source labelled dataset, which was used in [18] and made available on Open-Source Framework [19]. This corpus contains of legal cases including 58.16% taken from the United States supreme court. These cases are labelled into various categories like, Construction, Health Law, Family Law, Military and several more.

This paper uses 3 categories, namely, Military, Banking and Environment from the corpus for clustering of the legal documents belonging to these topics. Each of these categories have 218, 380, 881 documents respectively, totaling to 1479 documents.

4.2 Dataset Analysis

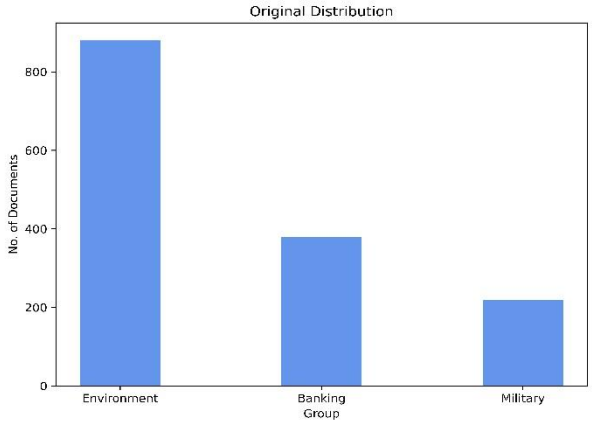

Fig. 2. Original Distribution of Documents

Fig.2 depicts the number of documents in the original 3 topic groups. As observed the original documents were non uniformly distributed with a certain group having clear dominance.

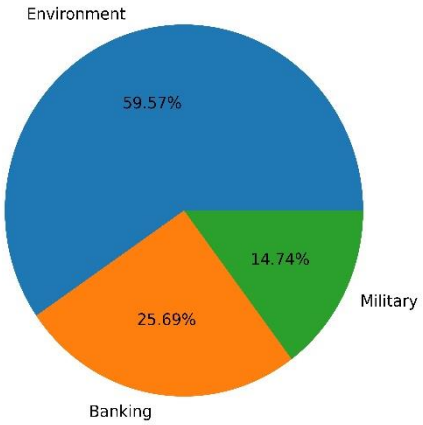

Fig. 3. Original Percentage Distribution of Documents

Fig.3 depicts the percentage distribution of the documents in the original 3 groups.

As visible in the above diagrams, the dataset was skewed and had a higher proportion of documents belonging to the environment group. The created clusters also have a

similar distribution. The competency of the model in such a skewed dataset implies that the model will have accurate responses to balanced datasets as well.

4.3 Comparative Modelling

The proposed methodology was tested against state-of-the-art approaches for document clustering. The python libraries, genism [20] and scikit-learn [21] were used for modelling. The proposed methodology involves Sentence Embeddings followed by UMAP and then Minibatch K-Means. The other models used are – Latent Dirichlet Allocation and BERTopic [14]. Another pipeline was tested which is, Sentence Embeddings followed by UMAP and then, clustering using K-Means. This pipeline was added in order to demonstrate the effectiveness of the Minibatch K-Means algorithm. Lastly, Sentence embeddings without UMAP was also tested. This helped to show the significance of dimensionality reduction on the results.

5 Results and Discussions

5.1 Performance Metrics

In order to evaluate the performance of the models used for the task of document clustering, certain performance metrics have been selected. Since the dataset provides labels denoting the topic of the documents, it enables the usage of certain performance metrics such as Purity, Precision, Recall and F1 score which are dependent on the presence of labels.

Since there are multiple topics which are being evaluated, there will be multiple values of precision, recall and F1-Score as they depend on the classes or topics in the data. In order to analyses the results better, the macro average of these metrics is compared. Macro average is selected in preference to macro weighted average due to the imbalance in the topics present in the data.

Table 1 shows the results recorded during experimentation with each model for Results for Purity, Homogeneity and Completeness. Table 2 shows the results for Precision, Recall and F1-score for each model.

Table 1. Results for Purity, Homogeneity and Completeness for each model

Model	Purity	Homogeneity	Completeness
DistilBert + MiniBatch K-Means	0.66	0.27	0.25
LDA	0.76	0.37	0.32
BERTopic	0.84	0.44	0.47
DistilBert + UMAP + K-means	0.77	0.53	0.48
DistilBert + UMAP + Mini batch K-Means	**0.85**	**0.53**	**0.49**

Table 2. Results for Precision, Recall and F1-score for each model

Model	Precision	Recall	F1-score
DistilBert + Mini batch K-means	0.48	0.27	0.40
LDA	0.19	0.28	0.21
BERTopic	0.62	0.65	0.67
DistilBert + UMAP + K-means	0.75	0.54	0.62
DistilBert + UMAP + Mini batch K-Means	**0.78**	**0.66**	**0.71**

5.2 Result analysis

On analyzing the performance of the models using the performance metrics mentioned, it can be observed that the sentence embeddings + UMAP + Minibatch K-Means architecture produces the best-case results and surpasses state-of-the-art models in the task of topic modelling and clustering documents according to the topic. The results indicate the importance of dimensionality reduction step on comparing the best-case results with the sentence embeddings + Minibatch K-Means architecture. It can also be observed that modifying the traditional K-Means clustering algorithm also provides better results. The best-case results can be further analyzed with the help of a confusion matrix which helps to understand the clusters better.

Figure 4 shows the confusion matrix of the results produced by the sentence embeddings + UMAP + Minibatch K-Means architecture.

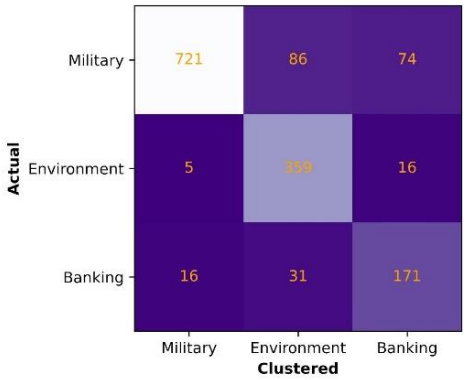

Fig. 4. Confusion Matrix of Proposed Model

As it can be seen, the confusion matrix shows the number of documents belonging to each topic and the actual and the clustered number of documents. It can be observed

that the topic ‘Military’ has 721 documents which belong to a single cluster which is around 81.83% of the documents belonging to the topic. Similarly, the topic ‘Environment’ has 359 documents and the topic ‘Banking’ has 171 documents belonging to their corresponding single clusters which is around 94.47% and 78.44% of the documents respectively. It can be concluded that the architecture performs well for all three topics. Along with the external measures for evaluation, internal metrics such as Purity, Homogeneity and Completeness show that the architecture produces clusters which are not very overlapping and distinctive as compared to the clusters formed by other architectures.

6 Conclusion and Future Work

The proposed model for legal document clustering successfully employs a context-aware approach with the use of the Mini Batch K-Means model. With the help of this model, topic modelling is carried out and the words with high influence in each topic indicate a logical relevance to each other in the legal context. The documents are categorized into clusters based on the topics outputted by the model. The performance of the system is encouraging because it produces results better than state-of-the-art topic modelling and clustering algorithms. Through the analysis of the corpus, it is evident that there lies an imbalance in dataset, implying that the system can even perform satisfactorily in such conditions.

Clustering of legal documents still continues to be difficult problem because of the diverse nature of these documents. There are several legal cases whose documentation is multi-topical in nature, insinuating that they could belong to more than one clusters. Future work can be done in researching on soft clustering algorithms and expanding the scope of legal document clustering to allow clusters to accurately belong to multiple clusters.

References

1. Venkatesh R kumar (2013) Legal Documents Clustering and Summarization using Hierarchical Latent Dirichlet Allocation. In: IAES International Journal of Artificial Intelligence (IJ-AI). Institute of Advanced Engineering and Science
2. Singh VK, Tiwari N, Garg S (2011) Document clustering using K-means, heuristic K-means and fuzzy C-means. In: Proceedings - 2011 International Conference on Computational Intelligence and Communication Systems, CICN 2011. pp 297–301
3. Mahdavi M, Abolhassani H (2009) Harmony K-means algorithm for document clustering. Data Mining and Knowledge Discovery 18:370–391. https://doi.org/10.1007/s10618-008-0123-0
4. Cui X, Potok TE, Palathingal P (2005) Document clustering using particle swarm optimization. In: Proceedings - 2005 IEEE Swarm Intelligence Symposium, SIS 2005. pp 191–197
5. Wu G, Fu E, Lin H, Wang L (2016) An Improved K-means Algorithm for Document Clustering. In: Proceedings - 2015 International Conference on Computer Science and

Mechanical Automation, CSMA 2015. Institute of Electrical and Electronics Engineers Inc., pp 65–69
6. Griffiths A, Robinson LA, Willett P (1984) Hierarchic agglomerative clustering methods for automatic document classification. Journal of Documentation 40:175–205
7. Conrad JG, Al-Kofahi K, Zhao Y, Karypis G (2005) Effective document clustering for large heterogeneous law firm collections. In: Proceedings of the International Conference on Artificial Intelligence and Law. ACM Press, New York, New York, USA, pp 177–187
8. Hotho A, Maedche A, Staab S (PDF) Ontology-based Text Document Clustering.
9. Svadas T, Jha J (2015) International Journal of Computer Science and Mobile Computing Document Cluster Mining on Text Documents. In: International Journal of Computer Science and Mobile Computing. pp 778–782
10. Diallo B, Hu J, Li T, et al (2021) Multi-view document clustering based on geometrical similarity measurement. International Journal of Machine Learning and Cybernetics 1–13. https://doi.org/10.1007/s13042-021-01295-8
11. Wagh RS, Anand D (2020) Legal document similarity: A multicriteria decision-making perspective. PeerJ Computer Science 2020:1–20. https://doi.org/10.7717/peerj-cs.26
12. Xie, P., Xing, E.P.: Integrating Document Clustering and Topic Modeling.
13. Angelov D TOP2VEC: DISTRIBUTED REPRESENTATIONS OF TOPICS
14. Grootendorst M, Reimers N (2021) MaartenGr/BERTopic: v0.9. https://doi.org/10.5281/ZENODO.5168575
15. Sanh V, Debut L, Chaumond J, Wolf T (2019) DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter
16. Mcinnes L, Healy J, Melville J (2020) UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction
17. Sculley D (2010) Web-scale k-means clustering. Proceedings of the 19th International Conference on World Wide Web, WWW '10 1177–1178. https://doi.org/10.1145/1772690.1772862
18. Sugathadasa K, Ayesha B, de Silva N, et al (2017) Synergistic Union of Word2Vec and Lexicon for Domain Specific Semantic Similarity
19. Silva N de, Ayesha B (2019) SigmaLaw - Large Legal Text Corpus and Word Embeddings. https://osf.io/qvg8s/. Accessed 11 Aug 2021
20. Rehurek R, Rehurek R, Sojka P (2010) Software Framework for Topic Modelling with Large Corpora. IN PROCEEDINGS OF THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS 45—50
21. Pedregosa F, Varoquaux G, Gramfort A, et al (2011) Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12:2825–2830