Topic Model for 20 NG Dataset using Clustering Algorithms

Aditya Singhal - 1154832 Hasib Kamal - 1155165

Abstract

With the exponential increase in the amount of content available on the web, internet users spend long hours finding relevant information. This project aims to explore a solution to help users spend less time on the searching process using clustering algorithms. We implement five different algorithms: K-Means, Latent Dirichlet Allocation (LDA), Non-Negative Matrix Factorization (NMF), Latent Semantic Indexing (LSI), and Hierarchical Dirichlet Processes (HDP) on a news documents dataset, and measure their performances using three extrinsic measures: Adjusted Random Index (ARI), Adjusted Mutual Information (AMI), and Normalized Mutual Information (NMI). Results indicate that NMF with TF-IDF is the most suitable clustering method for our use case, and we discuss some interesting findings.

Introduction

These days, one can find an endless amount of useful user-generated data on the internet that can help grasp new concepts. However, with the increase of scattered data, we are spending a lot of time finding the specific topic that we are looking for. Topic labeling and identifying attributes associated with text have always been a challenge since if this were to be done by human power, it would be time-consuming and inefficient.

Therefore, we decided to experiment with topic modeling. To put it simply, topic modeling processes a text (input) and automatically generates an appropriate topic (output) that is extracted through patterns found in the text corpus. By treating the instances as a mixture of topics, one can generate a cluster of keywords. With this, one can tell what the context of the text is by only reading its predicted topic, rather than reading the whole text, thus saving time when seeking information online. We evaluate several techniques for document embeddings and clustering on the 20 NG dataset. Our project implements clustering techniques as clustering is defined as a method of identifying similar groups of data in a corpus. It is a form of unsupervised learning useful for finding underlying patterns in the data.

The results obtained from this project can help in understanding the effectiveness of various clustering-based ML models on news websites having large amounts of unrelated news articles. This can aid experts to prioritize areas of study for tasks like controlling specific misinformation topics or analyzing public engagements on particular topics.

Related Work

20 Newsgroup dataset has been widely used for a variety of tasks. Previous research has involved text cleaning [1], feature selection [2], and distributional clustering [3].

Albishre, Khaled et al. [4] represented an effective way to clean the dataset using text cleaning pre-processing techniques. They also used text mining and feature selection to extract important features from the dataset to make it more compatible to be used in machine learning models.

Wang, P. Cui [5] proposed and implemented a structural deep network model called Structural Deep Network Embedding (SDNE) to perform a network embedding on the dataset which was able to map the data to a highly non-linear latent space and the model was able to preserve network structure representation. Bianchi, S. Terragni [6] proposed a simple method incorporating contextual embeddings into topic modeling models which generated better quality of topics and showed that context information is significant for topic modeling. Schubert, Lang, A., and Feher, G. [7] in their paper were able to avoid unnecessary similarity computation for spherical k-means clustering using triangle inequality for Cosine Similarity. They were able to speed up the algorithm significantly using this method.

Miao, Yu, & Blunsom, P. [8] introduced a deep neural variational inference framework text generative models where they experimented on two separate tasks, document modeling and question answering selection task for representing the performance of this framework. They used Neural Variational Document Model (NVDM) for document modeling and Neural Answering Selection Model (NASM) for evaluation. Their models achieved state-of-the-art performance for both cases. Wang, Rui, et al. [9] proposed a novel topic model named Bidirectional Adversarial Topic (BAT). They extended the BAT model so that it can parallelly work with word relatedness information and proposed the Bidirectional Adversarial Topic model with Gaussian (Gaussian-BAT). This model was able to outperform the state-of-the-art approaches in terms of coherence measures which is an intrinsic measure of the model's performance. Lin, Yuxiao, et al. [10] proposed BertGCN which is a combination Bert Model and Graph Convolutional Networks. This model uses both large-scale pretraining and transductive learning for text classification. It can construct a diverse graph for the corpus and uses GCN for classification. This model can be easily implemented on top of any document encoder as well as any graph model.

Dataset

20 Newsgroups[1] is a comprehensive dataset of newsgroup documents. It is a collection of 11,314 documents, referred to as instances, divided nearly evenly across 20 different topics. It consists of three feature columns: Text_id, Text, and Category. The dataset is divided into train-test (80:20) as we use the training part to find hyperparameters, and then evaluate clustering algorithms on the testing part. The average length of an instance is approximately 2,000 characters. Some of the instances are very closely related to each other (e.g. comp.sys.ibm.pc.hardware / comp.sys.mac.hardware), while others are highly unrelated (e.g misc.forsale/ soc.religion.christian). Table 1 shows a list of the 20 topics, separated according to the related subject.

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Table 1: List of topics grouped by their related subject

[1] https://www.kaggle.com/crawford/20-newsgroups

Approach and Methodology

Figure 1 shows an overview of the methodology followed in this project.

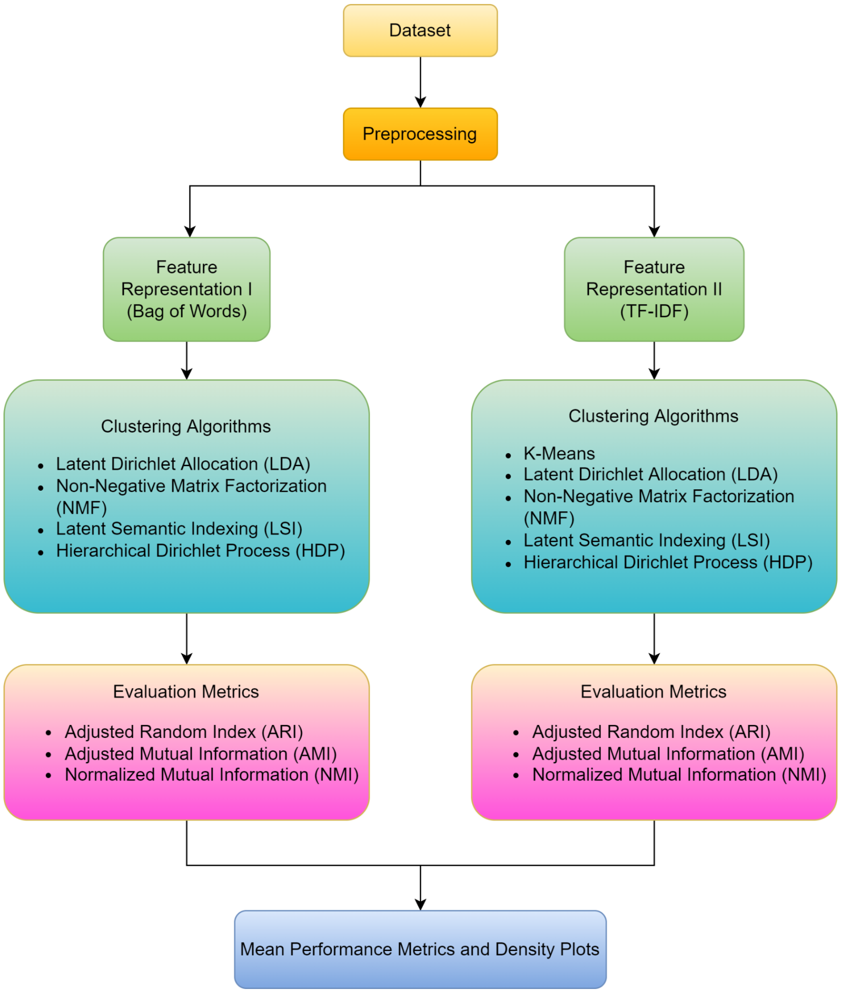
Dataset
Preprocessing
Feature Representation I (Bag of Words)
Feature Representation II (TF-IDF)
Clustering Algorithms
Latent Dirichlet Allocation (LDA)
Non-Negative Matrix Factorization (NMF)
Latent Semantic Indexing (LSI)
Hierarchical Dirichlet Process (HDP)
Clustering Algorithms
K-Means
Latent Dirichlet Allocation (LDA)
Non-Negative Matrix Factorization (NMF)
Latent Semantic Indexing (LSI)
Hierarchical Dirichlet Process (HDP)
Evaluation Metrics
Adjusted Random Index (ARI)
Adjusted Mutual Information (AMI)
Normalized Mutual Information (NMI)
Evaluation Metrics
Adjusted Random Index (ARI)
Adjusted Mutual Information (AMI)
Normalized Mutual Information (NMI)
Mean Performance Metrics and Density Plots

Fig 1. Flowchart outlining project methodology.

Preprocessing: First, the instances obtained from the dataset are cleaned. All instances are converted to lowercase, followed by the removal of all non-alphabets (punctuation, numbers, new-line characters and extra-spaces). Next, we remove all URLs and extra-spacing. Then English stopwords, using the online library nltk[2] were filtered out to remove low-level information from the claims to give more importance to the remaining words. Small words (length < 3) were removed as they cause complications and unreliability with the model's output. Stemming and lemmatization were also applied to the claims to reduce inflectional words and convert derivationally related forms of a word to a common base word to help with the model's output. To finish the pre-processing, tokenization was performed on the claims to break down the short texts into small chunks which are called tokens. Tokenization helps models by interpreting the meaning of the tokens by analyzing the sequence of the tokens.

[2] https://www.nltk.org/

Feature extraction: Features from the tokenized instances were extracted using two different feature representation techniques: Bag of Words (BoW) and Term Frequency – Inverse Document Frequency (TF-IDF).

A bag of Words (BoW) can be described as the occurrence of words within a document [11]. It is a simple representation method of extracting features from text which is widely used in machine learning algorithms models for topic modeling. It uses a very simple and flexible approach which can be used in different ways for extracting features from documents. It uses the intuition that documents are similar if they contain the same content. The reason it is called a 'bag' of words is simply that it does not put any emphasis on the structure of the words in the sentences. It only calculates whether the known words are occurring in the documents.

TF-IDF is a statistical measure that works by evaluating the relevance of a word in a document in a collection of documents [12]. It works by multiplying how many times a word appears in a sentence (Term Frequency) and the inverse frequency of the word across a collection of documents (Inverse Document Frequency).

Grid search was used to determine the optimal hyper-parameters for training the models. With the acquisition of optimal hyper-parameters, 5 topic modeling models were trained using the tokenized instances.

Clustering Algorithms: Next, we pass the features as input to various models from gensim library[3]. Latent Dirichlet Allocation (LDA) is a topic modeling model which labels topics based on word frequency from a set of documents [13]. It is used for discovering hidden grouping within documents and works well in large dataset. It gives multiple topics for each document and is more generalized in terms of topic generation. Latent Semantic Analysis (LSI) uses Singular Value Decomposition (SVD) a mathematical operation that reduces the input to its core for simple and efficient calculation [14]. Non-Negative Factorization (NMF) reduces the dimension of the input and uses the factor analysis method to give comparatively less weight to the words that have less coherence [15]. It is becoming popular for its ability to extract sparse and interpretable features. It works well with TF-IDF normalized documents. Hierarchical Dirichlet Process (HDP) is an extension of LDA which was designed to work where the number of topics in a document is not known. HDP uses the Dirichlet process to calculate the uncertain number of topics and learns the number of topics from the data. K-means is a distance-based algorithm that uses distances of the datapoints to calculate the centroid (center of the cluster) and form a cluster. Its main goal is to find datapoints that are similar and form a cluster. The main objective for a good cluster is to minimize the sum of the squared distance from each data point to its centroid. It's an iterative method that works by initializing a random centroid and recalculating it when data points are added to that cluster. K is the number of clusters which is a hyper-parameter (defined by the user).

Evaluation Metrics: As our dataset is labeled, we measure the performances by calculating three extrinsic measures: Adjusted Random Index (ARI), Adjusted Mutual Information (AMI), and Normalized Mutual Information (NMI). All vary between 0 and 1, and generally, the higher the score, the better a model is considered to perform.

[3] https://radimrehurek.com/gensim/

Density Plots: The seaborn[4] library was utilized to create statistical visual representations of density plots.

Results

Number of epochs is the key hyper-parameter for training a neural network. A large number of epochs can lead to over-fitting the data, however, a lower number of epochs can lead to under-fitting and bad performance. First, the performance change of the mean TF-IDF and Bag of Words (BoW) models was explored with the number of epochs. Results indicated the optimal number of epochs for TF-IDF as 175 and for BoW as 225.

The mean evaluation measures for the four feature representations with the clustering methods are discussed. Table 2 shows that using BoW as feature representation, LDA model gives the best performance and generates well-spaced clusters (Fig. 2). In addition, Table 3 indicates NMF as the best performing model on the feature extraction method TF-IDF.

Table 4 provides the mean for each of the three evaluation measures and the CPU time taken to train and test the models. Overall, NMF with TF-IDF generates the best results. The density plots shown in figure 3 confirm the same.

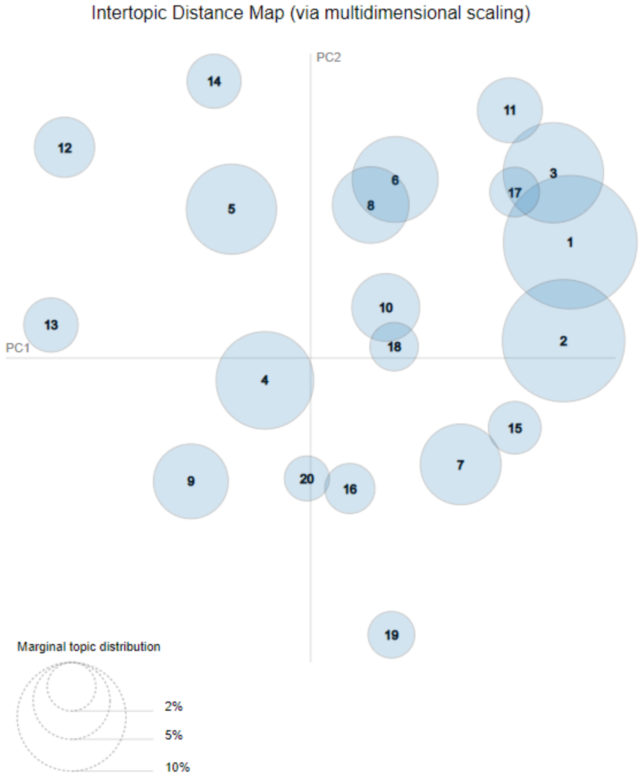

Fig 2. Intertopic distance map using LDA on Bag of Words

[4] https://seaborn.pydata.org/

Feature-extraction	**Clustering-algo**	**run#**	**state**	**AMI**	**ARI**	**NMI**	**time**
Bag of Words	NMF	1	2	0.238223	0.041038	0.264692	200.9551
Bag of Words	NMF	2	4	0.278979	0.069524	0.302633	201.3139
Bag of Words	**LDA**	**1**	**2**	**0.454268**	**0.280901**	**0.470963**	**1646.696**
Bag of Words	LSI	1	2	0.083412	0.004474	0.103052	7.678815
Bag of Words	LSI	2	4	0.083412	0.004474	0.103052	7.454431
Bag of Words	HDP	1	2	0.083412	0.004474	0.103052	71.59744
Bag of Words	HDP	2	4	0.083412	0.004474	0.103052	63.1918

Table 2: Performance Metrics using BoW as Feature extraction

Feature-extraction	**Clustering-algo**	**run#**	**state**	**AMI**	**ARI**	**NMI**	**time**
tf-idf	k-means	1	2	0.432945	0.152264	0.451518	24.14566
tf-idf	k-means	2	4	0.395782	0.113128	0.415687	13.98087
tf-idf	**NMF**	**1**	**2**	**0.467219**	**0.298985**	**0.482907**	**142.3671**
tf-idf	NMF	2	4	0.459631	0.290203	0.475536	130.4342
tf-idf	LDA	1	2	0.025723	0.000662	0.054395	1081.584
Bag of Words	LSI	1	2	0.089351	0.004145	0.120342	17.14844
Bag of Words	LSI	2	4	0.079562	0.00329	0.109348	15.26166
Bag of Words	HDP	1	2	0.079562	0.00329	0.109348	90.91767
Bag of Words	HDP	2	4	0.079562	0.00329	0.109348	86.95522

Table 3: Performance Metrics using TF-IDF as Feature extraction

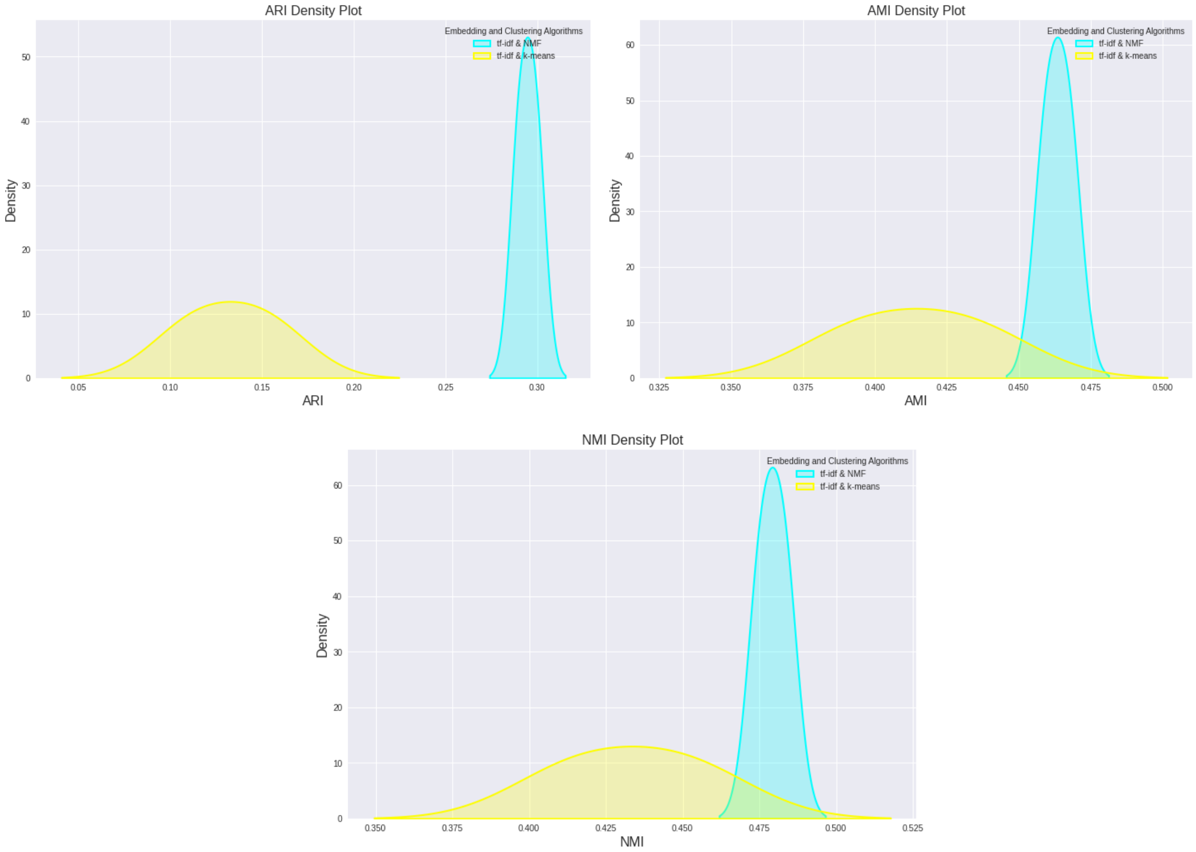

Fig 3. Density plots of the three extrinsic measures.

clustering-algo	ARI	AMI	NMI	time	embedding
HDP	0.002613	0.067591	0.095179	85.536283	TF-IDF
LDA	0.000662	0.025723	0.054395	1081.58418	TF-IDF
LSI	0.002488	0.064053	0.090843	13.654636	TF-IDF
NMF	**0.294594**	**0.463425**	**0.479222**	**136.400638**	**TF-IDF**
k-means	0.132696	0.414364	0.433602	19.063267	TF-IDF
HDP	0.004474	0.083412	0.103052	67.394618	Bag Of Words
LDA	0.280901	0.454268	0.470963	1646.69648	Bag Of Words
LSI	0.004474	0.083412	0.103052	7.566623	Bag Of Words
NMF	0.055281	0.258601	0.283663	201.134503	Bag Of Words

Table 4: Mean Performance Metrics

Conclusion and Discussion

We found NMF with feature extraction method TF-IDF to be the best performing model overall. Clustering using LSI takes the shortest time, while LDA takes the longest time. The results obtained from this project can help in understanding the effectiveness of various clustering-based ML models on news websites having large amounts of unrelated news articles. This can aid experts to prioritize areas of study for tasks like controlling specific misinformation topics or analyzing public engagements on particular topics.

Future work can include more feature extraction techniques. Researchers can also try implementing word embeddings pre-trained on a subset of data.

References

[1] Albishre, K., Albathan, M., & Li, Y. (2015, December). Effective 20 newsgroups dataset cleaning. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 3, pp. 98-101). IEEE.

[2] Liu, T., Liu, S., Chen, Z., & Ma, W. Y. (2003). An evaluation on feature selection for text clustering. In *Proceedings of the 20th international conference on machine learning (ICML-03)* (pp. 488-495).

[3] de Vries, E., M. Schoonvelde, and G. Schumacher. 2018. No Longer Lost in Translation: Evidence that Google Translate Works for Comparative Bag-of-Words Text Applications. Political analysis 26(4): 417–430. doi: 10.1017/pan.2018.26.

[4] K. Albishre, M. Albathan and Y. Li, "Effective 20 Newsgroups Dataset Cleaning," 2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), 2015, pp. 98-101, doi: 10.1109/WI-IAT.2015.90.

[5] Wang, D., P. Cui, and W. Zhu. 2016. Structural Deep Network Embedding. Proceedings of the 22nd ACM SIGKDD International Conference on knowledge discovery and data mining. ACM. p. 1225–1234

[6] Bianchi, F., S. Terragni, and D. Hovy. 2020. Pre-training is a Hot Topic: Contextualized Document Embeddings Improve Topic Coherence.

[7] Schubert, Lang, A., and Feher, G. (2021). Accelerating Spherical k-Means. https://doi.org/10.1007/978-3-030-89657-7_17

[8] Miao, Y., Yu, L., and Blunsom, P. (2016, June). Neural variational inference for text processing. In International conference on machine learning (pp. 1727-1736). PMLR.

[9] Wang, R., Hu, X., Zhou, D., He, Y., Xiong, Y., Ye, C., and Xu, H. (2020). Neural topic modeling with bidirectional adversarial training. arXiv preprint arXiv:2004.12331.

[10] Lin, Y., Meng, Y., Sun, X., Han, Q., Kuang, K., Li, J., and Wu, F. (2021). Bertgcn: Transductive text classification by combining gcn and bert. arXiv preprint arXiv:2105.05727.

[11] Zhao, and Mao, K. (2018). Fuzzy Bag-of-Words Model for Document Representation. IEEE Transactions on Fuzzy Systems, 26(2), 794–804. https://doi.org/10.1109/TFUZZ.2017.2690222

[12] Kim, and Gil, J.-M. (2019). Research paper classification systems based on TF-IDF and LDA schemes. Human-Centric Computing and Information Sciences, 9(1), 1–21. https://doi.org/10.1186/s13673-019-0192-7

[13] Jelodar, Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., and Zhao, L. (2018). Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. Multimedia Tools and Applications, 78(11), 15169–15211. https://doi.org/10.1007/s11042-018-6894-4

[14] Lossio-Ventura, Gonzales, S., Morzan, J., Alatrista-Salas, H., Hernandez-Boussard, T., and Bian, J. (2021). Evaluation of clustering and topic modeling methods over health-related tweets and emails. Artificial Intelligence in Medicine, 117, 102096–102096. https://doi.org/10.1016/j.artmed.2021.102096

[15] O'Callaghan, Greene, D., Carthy, J., & Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. Expert Systems with Applications, 42(13), 5645–5657. https://doi.org/10.1016/j.eswa.2015.02.055

[16] Altaweel, Bone, C., and Abrams, J. (2019). Documents as data: A content analysis and topic modeling approach for analyzing responses to ecological disturbances. Ecological Informatics, 51, 82–95. https://doi.org/10.1016/j.ecoinf.2019.02.014

[17] Alharbi, Hijji, M., & Aljaedi, A. (2021). Enhancing topic clustering for Arabic security news based on k‐means and topic modelling. IET Networks, 10(6), 278–294. https://doi.org/10.1049/ntw2.12017