Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 179 (2021) 40–46

Procedia Computer Science

www.elsevier.com/locate/procedia

5th International Conference on Computer Science and Computational Intelligence 2020

# Unsupervised News Topic Modelling with Doc2Vec and Spherical Clustering

Arif Budiarto[a,b,*], Reza Rahutomo[b,c], Hendra Novyantara Putra[a], Tjeng Wawan Cenggoro[a,b], Muhamad Fitra Kacamarga[a,d], Bens Pardamean[b,e]

[a] *Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480*
[b] *Bioinformatics and Data Science Research Center, Bina Nusantara University, Jakarta, Indonesia 11480*
[c] *Information System Department, School of Information System, Bina Nusantara University, Jakarta, Indonesia 11480*
[d] *Eureka AI, Singapore*
[e] *Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480*

**Abstract**

In the digital and Internet era, companies are racing to profile their target users based on their online activities. One of the reliable sources is the news articles they read that can represent their interests. However, extracting latent information from the news articles is not an easy task for a human. In this paper, we introduced a practical model to automatically extract latent information from news articles with pre-determined topics. Our proposed model used unsupervised learning, thus alleviating the need for humans to label news items manually. Doc2vec was used to generate word vectors for each article. Afterward, a spectral clustering algorithm was applied to group the data based on the similarity. A supervised Long Short Term Memory (LSTM) model was built to compare the clustering performance. The best 1, best 3, and best 5 scores were used to evaluate our model. The result showed that our model could not outperformed LSTM model for the best 1 score. However, the best 5 score result indicated that our model was sufficiently robust to cluster the articles based on topic similarity. Additionally, the proposed unsupervised model was implemented in both an on-premise server, and a cloud server. Surprisingly, our proposed method could run faster in the cloud server despite its less number of CPU cores.



*Keywords:* Topic Modelling; Latent Information; News Article; Document Embedding; Spherical Clustering;

* Corresponding author. Tel.: +62-812-1803-5608
*E-mail address:* abudiarto@binus.edu

1877-0509 
10.1016/j.procs.2021.12.007

## 1. Introduction

Exploring online activities from Internet users can yield useful information for building an online campaign or marketing[1,2,3]. Reading news article as a form of information fulfillment is one of the most popular online activities among Internet users[4]. Whether it is a sport, business, entertainment, or politics, each user has his/her unique interest. This latent information could be a valuable resource for companies to profile their target audiences, apart from only using demographic parameters such as age, gender, and location[5,6].

The process of extraction of this latent information from text data is part of the natural language processing (NLP) domain called topic modeling[7,8,9]. News topic modeling as a subset of NLP tasks is quite important to capture a person's interest just by looking at what s/he reads. The output of this task can be used to categorized readers based on their preference. Eventually, this categorization will be useful to create an effective targeted campaign. However, most of the time, a company will only get the unlabelled news article from several resources, make it hard to be inferred to a more useful insights. Herein, we describe a computational model for predicting news categories for Indonesian news articles acquired from several reputable news portals. We implement the model, provide benchmarking across two models on different computational platforms.

## 2. Related Works

### 2.1. Topical Concept Taxonomies

Our news topic model is closely related to a topical concept taxonomy model called TaxoGen[10]. TaxoGen is a model for topical concept taxonomies, one of NLP tasks that aims to generate hierarchical topics given a set of texts called corpus. In the paper, this task is implemented to generate hierarchical topics from a corpus contains scientific articles related to computer science. This particular task is different from topic modeling, whose goal is only to generate a topic given a document.

To generate topical concept taxonomy, TaxoGen utilizes word2vec[11]. This algorithm uses word vectors of the entire corpus as the basis for analysis. Our model uses doc2vec, which enables us to also incorporated document vector into the analysis. Thus, our model is capable of inferring document-level topics.

### 2.2. Word-Level Vector Embedding

In modern NLP, there is a tendency to embed word-level vectors into the model. In this type of embedding, each word in the corpus is assigned with a vector that uniquely represents the word. The numerical representation in the vectors is then learned through the optimization of a particular task. Relations between words is the main component being learned in this task. The first successful model of this embedding is word2vec[11]. word2vec has two types of optimization tasks to learn its vectors, skip-gram and CBOW (Continuous Bag-of-Words). In skip-gram, the model is asked to predict surrounding words given a word. On the opposite, CBOW predicts a word given its surrounding words.

Other commonly used word embedding models are fastText[12] and GLoVe[13]. FastText model is based on word2vec, but it breaks each word into sub-words to learn the structure of words. This allows the fastText model to be more robust for rare words than word2vec. Meanwhile, GloVe utilizes word-word co-occurrence for learning word vectors, thus it is faster than word2vec to be trained.

### 2.3. Document-Level Vector Embedding

In 2014, Le and Mikolov[14] extended word2vec to also learn document representation. This implementation is known as doc2vec. Alongside numerical representations of words, doc2vec will also include one vector that represents the document itself. With this treatment, doc2vec allows vector representation training for the documents and can be used to keep track of each document in further analysis.

| | link | sub_category | text | category |
|---|---|---|---|---|
| 0 | https://olahraga.kompas.com/read/2011/10/17/09... | Edukasi | Indonesian Corruption Watch menuntut Dinas Pe... | Edukasi |
| 1 | https://travel.kompas.com/read/2009/05/14/2106... | Sains | Untuk mendapatkan populasi orangutan di Kalima... | Sains |
| 2 | https://entertainment.kompas.com/read/2008/11/... | Sains | - Direktur Jenderal Sejarah dan Purbakala Dep... | Sains |
| 3 | https://entertainment.kompas.com/read/2008/09/... | Oase | --Masyarakat Kristen Indonesia di Yogyakarta, ... | Wisata |
| 4 | https://sains.kompas.com/read/2010/05/18/23143... | Liga Italia | - Pelatih Inter Milan, Jose Mourinho, semakin ... | Olahraga |
| 5 | https://tekno.kompas.com/read/2013/03/16/15121... | Gadget | , - Sejak diluncurkan pada akhir tahun 2012 l... | Teknologi |
| 6 | https://sains.kompas.com/read/2010/08/16/16240... | Travel | - Sehari sebelum menyambut bulan ramadhan, ka... | Wisata |
| 7 | https://olahraga.kompas.com/read/2009/08/31/09... | Properti | KRISIS keuangan global membuat sebagian orang ... | Properti |
| 8 | https://travel.kompas.com/read/2008/09/11/1142... | Perempuan | ,1 ekor ayam, belah dadanya tanpa terputus, t... | Lifestyle |
| 9 | https://olahraga.kompas.com/read/2011/05/20/09... | News & Features | Hampir setiap kaum hawa pasti mengidamkan kuli... | Kesehatan |

Fig. 1: Data Structures and the First 10 Rows Data

## 3. Material and Methods

### *3.1. Dataset*

All articles used in this research were collected from several reputable news portals such as liputan6.com, kompas.com, and detik.com. A web scraping tool was built using Scrapy to collect the articles from these sources [15]. In total there were more than 1 million articles collected during a week of web scraping, at the end of June 2018. During the web scraping process, metadata including Uniform Resource Locator (URL), title, date, author, and category were captured alongside the text of the main content. The detail level of the categories captured directly from the sources was varied. The majority of them were categorized at a very low level of detail. For instance, within the football section, the articles were categorized into more detailed classes such as Liga Italia, Liga Primer Inggris, etc. Because of this fact, the parent class was defined based on the hierarchical structure from the sources. There were more than 200 categories captured directly from the sources. After the mapping process this number could be reduced to only 11 main categories to classify all the articles: 'Ekonomi' (economics), 'Hiburan' (entertainment),'Kesehatan' (health), 'Lifestyle', 'Otomotif' (automotive), 'Pendidikan' (education), 'Properti' (property), 'Sains' (science), 'Teknologi' (technology), 'Wisata' (tourism), 'Sepakbola' (football).

Moreover, the distribution of articles across categories was not equal. For popular categories such as football or economics, the number of articles reached more than 10,000. On the other hand, there were about 4,000 articles for less popular categories such as education. Knowing this fact, a stratified sampling approach was applied and consequently filtered down the total number of articles to 40,000. Finally, after the process of data cleansing including stop words and punctuation removal and to meet memory limitations during the data analysis process, this number was further reduced to nearly half by doing the stratified sampling based on the categories. In the end, 28,402 articles remained in the analysis, with 2,582 articles for each main category. The data structure and the first 10 rows of the data can be seen in Figure 1.

### *3.2. Methods*

Our method was based on TaxoGen [10] model. To enable TaxoGen to generate the topic for each document, we used doc2vec instead of word2vec for the embedding model. The embedding model allowed us to generate vector representations for each document. These document vectors were then clustered by spherical clustering [16], as what was done in TaxoGen for all words in its corpus. Spherical clustering is an alternative to standard k-Means clustering. Instead of using euclidean distance to clusters all data points, this method focus on the angle measurement between vectors, commonly known as a cosine similarity score. We used 11 clusters to match the number of categories in our data.

Afterward, our proposed model checked for the nearest vector of topic words to all documents in each cluster. The topic words were limited to the pre-defined categories we mentioned before. We treated the nearest topic words as predicted classes of all documents in the corresponding cluster.

To measure this proposed methodology performance, best 1, best 3 and best 5 scores were used to determine the accuracy of the model. For best 1 score, the predicted class were compared against the actual label generated from

the news sources as the ground truth. While for the best 3 and best 5 scores, the ground truth was compared to the top 3 and top 5 predicted labels. The last 2 scores could give another perspective which shows the relation of one article with multiple topics. This proposed method was implemented in two different environments with the specification listed in Table 1.

Table 1: Hardware Architecture Comparison

| Environment | CPU | Memory | GPU |
|---|---|---|---|
| Server | 2x Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20 GHz (40 cores) | 86 GB | NVIDIA Tesla P100 |
| Cloud- Amazon Web Service (AWS) | EC2 C5.4xlarge - Intel Xeon Platinum @ 3.0 GHz (16 cores) | 32 GB | - |

### *3.3. LSTM for Comparison*

We built one variant of Recurrent Neural Network (RNN) named Long Short Term Memory (LSTM) [17] to compare the output from spherical clustering. LSTM was chosen as its ability to learn dependencies in the sequence of higher-level features [18] [19]. The model worked by fitting our features which translated into NumpyArray to our label which encoded into one hot encoder. We split the data into two groups: training dataset and test dataset. The training data contained 17,000 records and the remaining was a test dataset.

The architecture of this LSTM model was relatively simple. This model was started by embedding layer with the size of 20, correspond the the vector size generated by doc2vec. It is followed by an LSTM layer to learn the characteristic of the sequence features and followed by a Dense layer in the end as the output layer which generate the predicted label from 11 classes. The model architecture finished with the compilation of all layers with binary cross-entropy as the loss function.

Because LSTM is a supervised learning method, we need to label the clusters generated by our method for a direct comparison. In this study, we labeled a cluster with the majority of training data class in the cluster. Thus, the accuracy of our method can be calculated from the label of the nearest cluster of the test dataset.

In addition to the aforementioned accuracy calculation, we also inspected the accuracy of our method based on the top 3 and top 5 classes predicted by our method. The top 3 means that in the accuracy calculation, the label nearest 3 clusters are considered for the prediction, which the prediction is determined as true if one of the three cluster labels matched the ground truth label. By this definition, the first accuracy calculation can be called as top 1. The top 1 and 5 prediction is common for assessing classification performance, for instance, in image classification [20,21,22]. The top 3 was also inspected to smooth the analysis of the result.

## 4. Results and Discussion

Our proposed method generated two main important outcomes. The first one was the keywords for each group from the clustering model as can be seen in Table 2. The second outcome was the inference result for each document based on the pretrained model. Table 3 shows the first 5 rows of this result including the predicted cluster and its corresponding confidence level.

To evaluate the performance of our methodology, best 1 score, best 3 score, and best 5 score were used with the results are 39.75%, 69.35% and 87.30% respectively. The best 1 score indicates that our methodology is not sufficiently robust to predict the actual class compared to the LSTM model which can reach 95% accuracy.

However, In Table 4, we can see that the true positive (TP) for each category is increased from the best 1 to the best 5 score. For 'Ekonomi', 'Hiburan', 'Kesehatan', and 'Otomotif' category the increase is significant from 0% accuracy in best 1 score to almost 90% accuracy in best 5 score for overall.

A relatively good best 5 score for overall categories is an interesting point to be highlighted. One factor that causes this result is because one news article can belong to multiple categories regarding the broad topics of this article. For instance, if we see the similarity scores in cluster 6, as can be seen in Table 5, it is clear that the difference of similarity scores for the top 5 predicted topics is quite narrow. In other words, we can say that the articles belong to cluster 6 can be considered to have 5 topics with similar relevancy. This finding strongly suggest to take account this unique

Table 2: Cluster Keywords

| **Cluster0** | **Cluster1** | **Cluster2** | **Cluster3** | **Cluster4** | **Cluster5** | **Cluster6** | **Cluster7** | **Cluster8** | **Cluster9** | **Cluster10** |
|---|---|---|---|---|---|---|---|---|---|---|
| indonesia | tahun | makanan | mobil | anda | pengguna | sekolah | persen | warna | itu | rumah |
| wisata | itu | penyakit | produk | anak | aplikasi | pendidikan | rp | rumah | pemain | tahun |
| kota | hutan | kesehatan | unit | pria | google | indonesia | tahun | ini | ini | perumahan |
| tahun | warga | air | motor | pasangan | internet | guru | harga | ruang | saya | jakarta |
| hotel | gunung | obat | ini | perempuan | situs | siswa | bank | anda | film | properti |

Table 3: Inference Result

| Document_No | Dominant_Topic/ Cluster | Confidence_Level | Text |
|---|---|---|---|
| 0 | 6 | 0.985099971 | ['indonesian', 'corruption', 'watch', 'menuntut', 'dinas', 'pendidikan', . . .] |
| 1 | 1 | 0.773199975 | ['untuk', 'populasi', 'orangutan', 'kalimantan', 'akurat', 'survei', . . .] |
| 2 | 1 | 0.627600014 | ['direktur', 'jenderal', 'sejarah', 'purbakala', 'departemen', 'kebudayaan', . . .] |
| 3 | 6 | 0.716300011 | ['masyarakat', 'kristen', 'indonesia', 'yogyakarta', 'sabtu', 'menemui', . . .] |
| 4 | 9 | 0.750500023 | ['pelatih', 'inter', 'milan', 'jose', 'mourinho', 'kerasan', 'italia', . . .] |
| 5 | 10 | 0.590200007 | ['sejak', 'diluncurkan', 'tahun', '2012', 'lalu', 'microsoft', 'angka', . . .] |
| 6 | 8 | 0.52759999 | ['sehari', 'menyambut', 'ramadhan', 'sekeluarga', 'liburan', 'musim', . . .] |

Table 4: True Positive for Each Category based on 3 Scores

| Category | Best 1 TP | Best 3 TP | Best 5 TP |
|---|---|---|---|
| Ekonomi | 0 | 1795 | 1954 |
| Hiburan | 0 | 2379 | 2379 |
| Kesehatan | 0 | 1968 | 2432 |
| Lifestyle | 580 | 648 | 2328 |
| Otomotif | 0 | 1923 | 2489 |
| Pendidikan | 2386 | 2408 | 2428 |
| Properti | 1731 | 1774 | 1774 |
| Sains | 134 | 236 | 2287 |
| Sepakbola | 2430 | 2447 | 2447 |
| Teknologi | 2227 | 2256 | 2256 |
| Wisata | 1803 | 1865 | 2021 |
| **Accuracy** | **39.75%** | **69.35%** | **87.30%** |

characteristics of a news article in developing the clustering method. One of the directions is to implement more robust embedding method[23,24] and combined it with hierarchical clustering method[25,26,27].

Table 5: Top 5 Similarity Scores in Cluster 6

| Category | Similarity Scores |
|---|---|
| otomotif | 0.182413 |
| sains | 0.197086 |
| kesehatan | 0.200201 |
| teknologi | 0.244351 |
| wisata | 0.258912 |

Another point can be reported from this research is the comparison of execution time in 2 different environments. We implemented our methodology in an on-premise server and cloud environment. We split the whole methodology into three main parts: pre-processing, doc2vec modeling and spherical clustering. We can see from Table 6 that our methodology ran faster in the cloud environment than the on-premise server despite the different number of CPU

cores. This result indicates that our proposed methodology does not optimize parallel processing through all CPU cores.

Table 6: Execution Time Comparison for 2 Environments (in seconds)

| Task | On-premise Server | AWS |
|---|---|---|
| Pre-processing | 75 | 68 |
| Doc2Vec | 3930 | 2737 |
| Spherical Clustering | 214 | 88 |
| **TOTAL** | **4219** | **2893** |

## 5. Conclusion

We introduced a combination of doc2vec and spherical clustering as an unsupervised model for Indonesian news topic modeling. Alongside the methodology, a dataset contains 28,402 articles in 11 main categories was also provided. This kind of research was still lacking for the Indonesian language particularly, especially in unsupervised learning case which does not require the labeled dataset. Our research could be a good foundation for further research with several directions including hyper parameter tuning strategy, building a more solid clustering algorithm, and also generating hierarchical topic modeling. We also provided an HPC point of view by implementing and comparing the execution time of our methodology in 2 different environments. This result suggests us to do more research on developing an algorithm that can take benefits of parallel computing and the use of the Graphical Processing Unit (GPU).

## 6. Acknowledgment

This research is a collaborative effort between Eureka AI and Computer Science Department, Bina Nusantara University. The high-performance computer (HPC) resources in this research used NVIDIA Tesla P100 provided by NVIDIA Institute-Artificial Intelligent Research and Development Center, Bina Nusantara University. The cloud computing service is provided by Amazon Web Service (AWS) Cloud Credits for Research grant.

## References

1. Gasparini, M., Nuara, A., Trovò, F., Gatti, N., Restelli, M.. Targeting optimization for internet advertising by learning from logged bandit feedback. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. 2018, p. 1–8.
2. Farahat, A., Bailey, M.C.. How effective is targeted advertising? In: *Proceedings of the 21st International Conference on World Wide Web*; WWW '12. New York, NY, USA: Association for Computing Machinery. ISBN 9781450312295; 2012, p. 111–120. doi:\bibinfo{doi}{10.1145/2187836.2187852}.
3. Fong, N.. Targeted marketing and customer search. *ACR North American Advances* 2012;.
4. Szabo, A., Allen, J., Stephens, C., Alpass, F.. Longitudinal Analysis of the Relationship Between Purposes of Internet Use and Well-being Among Older Adults. *The Gerontologist* 2018;**59**(1):58–68. doi:\bibinfo{doi}{10.1093/geront/gny036}.
5. Tomšů, R., Marchal, S., Asokan, N.. Profiling users by modeling web transactions. In: *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. 2017, p. 2399–2404.
6. Kacamarga, M.F., Cenggoro, T.W., Budiarto, A., Rahutomo, R., Pardamean, B.. Analysis of acoustic features in gender identification model for english and bahasa indonesia telephone speeches. *Procedia Computer Science* 2019;**157**:199–204.
7. Ghosh, S., Chakraborty, P., Nsoesie, E.O., Cohn, E., Mekaru, S.R., Brownstein, J.S., et al. Temporal topic modeling to assess associations between news trends and infectious disease outbreaks. *Scientific reports* 2017;**7**(1):1–12.
8. Moon, S., Chung, S., Chi, S.. Topic modeling of news article about international construction market using latent dirichlet allocation. *Journal of The Korean Society of Civil Engineers* 2018;**38**(4):595–599.
9. Liu, D.R., Liao, Y.S., Lu, J.Y.. Online news recommendations based on topic modeling and online interest adjustment. *Industrial Management & Data Systems* 2019;.
10. Zhang, C., Tao, F., Chen, X., Shen, J., Jiang, M., Sadler, B., et al. TaxoGen: Unsupervised Topic Taxonomy Construction by Adaptive Term Embedding and Clustering. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*

- *KDD '18*; 2. New York, New York, USA: ACM Press. ISBN 9781450355520; 2018, p. 2701–2709. doi:\bibinfo{doi}{10.1145/3219819.3220064}.
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.. Distributed Representations of Words and Phrases and their Compositionality. In: *Proceedings of the 26th International Conference on Neural Information Processing System*; vol. 2. 2013, p. 3111–3119. 1310.4546.
12. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 2017;**5**:135–146.
13. Pennington, J., Socher, R., Manning, C.. Glove: Global vectors for word representation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, p. 1532–1543.
14. Le, Q., Mikolov, T.. Distributed representations of sentences and documents. In: *International Conference on Machine Learning*. 2014, p. 1188–1196.
15. Scrapy, A.. Fast and powerful scraping and web crawling framework. 2016.
16. Dhillon, I.S., Modha, D.S.. Concept decompositions for large sparse text data using clustering. *Machine Learning* 2001;**42**(1-2):143–175. doi:\bibinfo{doi}{10.1023/A:1007612920971}.
17. Hochreiter, S., Schmidhuber, J.. Long short-term memory. *Neural computation* 1997;**9**(8):1735–1780.
18. Zhang, X., Zhao, J., LeCun, Y.. Character-level convolutional networks for text classification. In: *Advances in neural information processing systems*. 2015, p. 649–657.
19. Zhou, C., Sun, C., Liu, Z., Lau, F.. A c-lstm neural network for text classification. *arXiv preprint arXiv:151108630* 2015;.
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. 2012, p. 1097–1105.
21. Simonyan, K., Zisserman, A.. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:14091556* 2014;.
22. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, p. 1–9.
23. Li, C., Duan, Y., Wang, H., Zhang, Z., Sun, A., Ma, Z.. Enhancing topic modeling for short texts with auxiliary word embeddings. *ACM Transactions on Information Systems (TOIS)* 2017;**36**(2):1–30.
24. Li, D., Zamani, S., Zhang, J., Li, P.. Integration of knowledge graph embedding into topic modeling with hierarchical dirichlet process. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, p. 940–950.
25. Kim, H., Drake, B., Endert, A., Park, H.. Architext: Interactive hierarchical topic modeling. *IEEE Transactions on Visualization and Computer Graphics* 2020;.
26. Yang, Y., Yao, Q., Qu, H.. Vistopic: A visual analytics system for making sense of large document collections using hierarchical topic modeling. *Visual Informatics* 2017;**1**(1):40–47.
27. Yu, D., Xu, D., Wang, D., Ni, Z.. Hierarchical topic modeling of twitter data for online analytical processing. *IEEE Access* 2019;**7**:12373–12385.