# Short News Text Clustering
(Information Retrieval and Text Mining Term Project Report)

Ans Munir
Information Technology University
Lahore
msds20033@itu.edu.pk

## Abstract

*Conventional methods such as Bag-of-word representation or TF-IDF representation cannot be used to cluster short texts since the resulting sparse vector representation will be unsatisfactory. To cluster short text, two methods or models have been put forth. Using Google's pre-trained Word2Vec model, the original text is integrated in the first model as compact binary codes. Convolutional Neural Networks (CNNs) are then fed these embeddings to obtain dense feature representations. The pre-trained binary codes are then fitted using the CNN's output units throughout the training process. Finally, the learned representation has been clustered using k-Means in order to get the optimal number of clusters. In the second model, the auto-encoder and sentence embedding are used to train the model discriminative features. Subsequently, it employs assignments derived from the clustering process and monitors the encoder network's weight updates.*

## 1. Introduction

Text clustering is the process of automatically grouping similar texts together without the need for outside supervision. Usually we have effective methods for grouping large texts. We've had a lot of brief text produced recently by news websites, social media, and other sources.

This short text is very important, and there are many uses for it, including recommendation and social media text clustering, among many others. The sparsity issue emerges when we employ conventional methods like the bag-of-word approach or Term Frequency-Inverse Document Frequency (TF-IDF) vectors since most words only occur once. As a result, the usefulness of these conventional similarity metrics declines.

Low-dimensional short text representations have recently demonstrated promise in resolving the sparsity issue in short text clustering. In various machine learning for natural language processing tasks, embeddings of words, phrases, and documents have been shown to be beneficial when paired with neural network design.

Neural Networks, such as Recursive Neural Networks (RecNN) and Recurrent Neural Networks (RNN), perform exceptionally well when constructing text representation through word embedding. However, their performance is not particularly impressive. RecNN's excessive temporal complexity during textual tree construction is the source of its difficulty. However, because the words that appear later are more supreme than the words that come before them, RNNs that use hidden layers are biassed models.

CNN has emerged as the most well-known and unbiased model in recent years. By using convolutional filters, it obtains local characteristics. As a result, CNN has improved its performance on several NLP tasks, such as relation categorization and sentence modelling.

## 2. Literature Review

The sparseness issue with brief text clustering has been the subject of numerous investigations. Expanding and improving the data's content is one technique. Banerjee et al. [1] provide one illustration of the aforementioned technique. He gave a demonstration of how to improve short text clustering by adding more features to its representation. He made use of the Wikipedia text. The work of Fodeh et al. [3] would be another illustration. He incorporates ontology semantic knowledge into the text grouping. However, these approaches haven't worked out since they require a deep understanding of natural language processing and continue to employ high-dimensional representation, which can waste computational time and memory.

An additional technique that has been used is mapping the original features onto the smaller space. Locality Preserving Indexing (LPI) [4], Laplacian Eigenmaps (LE) [7], and Latent Semantic Analysis (LSA) [2] are a few of its examples. Additionally, very sophisticated models for grouping brief texts have been examined by researchers. Similar to Tang et al. [9] novel architecture, which uses matrix factorization to simultaneously lessen real features and enhance text features through machine translation. The majority of the aforementioned approaches ignore word order in texts and are part of superficial structures, making it difficult to determine precise semantic similarities.

Deep Neural Networks (DNN) have been widely employed in research recently for feature learning. Identical to how Salakhutdinov and Hinton [5] employed DAE to teach text representation. During the fine-tuning procedure, they employed backpropagation to obtain codes that are useful for re-establishing the word-count vector.

Word-embedding, or learning a distributed representation of every word, has been proposed by researchers recently, and it involves using an external corpus. Deep Neural Network's performance on NLP tasks will improve as a result. Word2Vec and Skip-gram are two such models. A more recent model, called GloVe, was introduced by Pennington et al. [8] and depicts the global corpus statistics.

Le and Mikolov [6] extended the previous Word2Vec model by forcasting words in sentences in order to learn dense representation vectors of the text sentences; this new model is called Paragraph Vector (Para2vec). It's still a superficial, window-based approach. Large corpus is necessary for it to perform well. Word embedding is a common technique used by neural networks to accurately capture important syntactic and semantic regularities. These consist of RNN and RecNN. However, they also have some issues. Similar to RecNN, which builds textual trees with a high temporal complexity, RNN is a biassed model since the layer it uses to represent the text is computed at the final word.

Due to its effectiveness in learning implicit characteristics without bias, CNN has been used extensively for a variety of supervised Natural Language Processing applications. More recently, Visin et al. [10] attempted to use four RNNs known as ReNet to replace the CNN's convolutional layer in order to learn features that are not biassed for object identification. ReNet sweeps in many directions—from top to bottom, bottom to top, right to left, and left to right—over lower-layer characteristics.

## 3. Dataset

Our data set was obtained from https://opendata.com.pk. This data collection is made up of the news title, news paragraph, and label. You may access the data set at this Link

It is released by the ITU's Crime Investigation and Prevention Lab (CIPL). The data set includes both domestic and international crime news. The following are the many news categories: police, rape, robbery, shooting, theft, torture, arrest, and kidnapping; burglary, burn; cheating; child abuse; dacoity; extortion; fraud; harass; kill; and murder.

Physically we have one data set but virtually we have two data sets because we have also used news titles in our project and also cluster it through our models. As our project is about clustering short text therefore news titles are also used for clustering.

## 4. Methodology

1. First Model:

A framework for self-taught learning has been used in the first model, which is a self-taught convolutional neural network. In particular, the original text's raw text features are integrated into little binary codes. The pretrained Word2vec embeddings from Google have been utilised to translate text into binary codes. These binary codes have undergone dimensionality reduction using a traditional unsupervised approach. Next, text matrices are input into a CNN model to learn how to represent deep features through word embeddings. Output units are used to help fit pretrained binary codes. We use the K-means algorithm to process the learnt features and group short text into the appropriate clusters.

2. Second Model:

There are three primary steps in the second model. (1) SIF embeddings are used to embed short text. (2) The short text SIF embeddngs are encoded and reconstructed using a deep autoencoder in a pre-training phase. (3) The soft-cluster assignment technique has been applied as an auxiliary target distribution during the self-training phase. Aditionally, it simultaneously refines the clustering assignments and encoder weights.

The first phase involves using Smooth Inverse Frequency (SIF) embeddings to embed the text. The second stage involves initialising the encoder network's settings using a deep autoencoder architecture. Self-training is the focus of the third step. The K-means clustering method has been applied in this phase.

## 5. Results

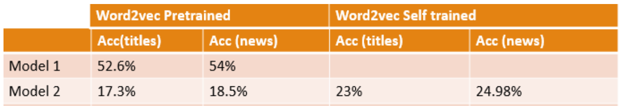
1. First Model:

We have used Google's pretrained Word2vec embeddings. We have used two data sets. First is news titles and second is news detail paragraph. On news titles we got accuracy of 52.6%. On news detail paragraph we got accuracy of 54% as shown in Table 1.

Then we have checked for Normalized Mutual Information (NMI). NMI for news titles data set is 18% and NMI for news paragraph data set is 22.1% as shown in Table 2.

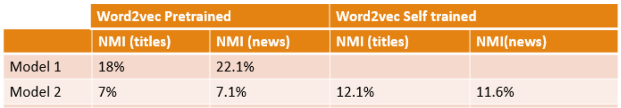
2. Second Model:

Two different kinds of embeddings are employed in the second model. At first, we employed the pretrained word2vec embeddings from Google. After that, we made use of our own Word2vec pretrained model.



| | Word2vec Pretrained | | Word2vec Self trained | |
|---|---|---|---|---|
| | Acc(titles) | Acc (news) | Acc (titles) | Acc (news) |
| Model 1 | 52.6% | 54% | | |
| Model 2 | 17.3% | 18.5% | 23% | 24.98% |

Table 1. Accuracy for model 1 and model 2 for news titles data set and news paragraph data set



| | Word2vec Pretrained | | Word2vec Self trained | |
|---|---|---|---|---|
| | NMI (titles) | NMI (news) | NMI (titles) | NMI(news) |
| Model 1 | 18% | 22.1% | | |
| Model 2 | 7% | 7.1% | 12.1% | 11.6% |

Table 2. NMI for model 1 and model 2 for news titles data set and news paragraph data set

As indicated in Table 1, we obtained 17.3% accuracy on the news titles data set and 18.5% accuracy on the news paragraph data set when we employed a pretrained model. As table 2 illustrates, we obtained 7.1% NMI on the news headlines data set and 7.1% NMI on the news paragraph data set.

Next, we used our own data set to train the word2vec model. As seen by Table 1, we have 23% accuracy on the news titles data set and 24.98% accuracy on the news paragraph data set. As indicated in table 2, we also obtained 12.1% NMI for the news headlines data set and 11.6% NMI for the news paragraph data set.

In the second model, it is evident that training our own data set resulted in improvements in accuracy and NMI when compared to utilising Google's pretrained Word2vec model.

## 6. Conclusion

Short text clustering differs from long text clustering. Therefore, we are unable to use the conventional clustering algorithms that we employ for long text to short text. Convolutional Neural Networks, however, could successfully cluster brief text. CNN has been utilised in both models to cluster brief text.

## References

[1] Somnath Banerjee, Krishnan Ramanathan, and Ajay Gupta. Clustering short texts using wikipedia. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 787–788, 2007. 1

[2] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990. 1

[3] Samah Fodeh, Bill Punch, and Pang-Ning Tan. On ontology-driven document clustering using core semantic features. *Knowledge and information systems*, 28:395–421, 2011. 1

[4] Xiaofei He and Partha Niyogi. Locality preserving projections. *Advances in neural information processing systems*, 16, 2003. 1

[5] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006. 2

[6] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR, 2014. 2

[7] Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14, 2001. 1

[8] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 2

[9] Jiliang Tang, Xufei Wang, Huiji Gao, Xia Hu, and Huan Liu. Enriching short text representation in microblog for clustering. *Frontiers of Computer Science*, 6:88–101, 2012. 2

[10] Francesco Visin, Kyle Kastner, Kyunghyun Cho, Matteo Matteucci, Aaron Courville, and Yoshua Bengio. Renet: A recurrent neural network based alternative to convolutional networks. *arXiv preprint arXiv:1505.00393*, 2015. 2