Analysis of Text Clustering Using Large Pre-trained Models

Zhiyu Chen, Wenxin Feng, Maia Guo, Shu Wang
{zc987, wf541, yg2483, sw5034}@nyu.edu
Natural Language Understanding, New York University
Github: https://github.com/Cheryl008/NLU_project.git

Abstract

Large pre-trained models have helped researchers to extend state-of-the-art for natural language understanding tasks. Recent papers have shown that combining large pre-trained language models with classic clustering algorithms can greatly improve text clustering performance as these models capture both basic linguistic patterns and generalized information from the corpus. In this paper, we design a 2-stage experiment to explore the impact of different language representation models on text clustering performance. Results prove that large pre-trained models can significantly improve the performance of text clustering algorithms.

1 Introduction

Text clustering is one of the fundamental tasks of Natural Language Processing (NLP) and has a huge commercial value as it plays a crucial role in numerous services in our daily lives, such as personalized advertising (Huang et al., 2021), news recommendation (Bouras and Tsogkas, 2017), spam detection (Sasaki and Shinnou, 2005), and social media sentiment analysis (Kharlamov et al., 2019). However, the strong dependency on labeled data largely prevents its further application in new settings, as unstructured, unlabeled text data is the most common type of data in explosively-growing real-world information (Chen et al., 2020). Obtaining sufficient labeled data through traditional manual annotation is time-consuming and expensive (Lee and Yang, 2009). Therefore, clustering text with unlabeled data, which can significantly reduce human resource costs, improve system efficiency (Tao et al.), and use the nature data to its full potential, is a highly promising and valuable study direction.

Early text clustering studies focus on statistical methods of feature selection (Yang and Pedersen, 1997; Houle and Grira, 2007) and apply basic classifiers, like TF-IDF, to realize the categorization task. With the booming development of Large pre-trained language models in recent years, researchers began to leverage BERT (Devlin et al., 2019) and its variant as a shared feature extractor. It has been proven outstanding clustering performance by combining BERTs with traditional clustering algorithms (Guan et al., 2020; Huang et al., 2020; Stammbach and Ash, 2021; Shnarch et al., 2022). Although the Few-Shot models are more popular in many fields today, they still need some human labeled data to do semi-supervised learning. As long as human work is involved, the problems of subjectivity and domain limitations will always exist (Yao, 2021). Therefore, regarding unsupervised text clustering tasks, large pre-trained language models are more worthy of investigation. However, the related researches are still limited, and especially there are some disconnects between the combinations of the representation models, the clustering algorithms, as well as the designed evaluation metrics. Our research will focus on filling these experimental gaps to explore the best combinations and to demonstrate the advantages of pre-trained language models.

From the experiments, we find that large pre-trained language models such as ELECTRA and DeBERTa significantly improve the clustering performance compared to baseline (TF-IDF), especially after fine-tuning on training set. A fine-tuned DeBERTa or ELECTRA can help most clustering algorithm to achieve almost 98.6% accuracy on R2 dataste, with NMI score around 0.89 and ARI score around 0.95. Notably, while pre-trained language models excels in dataset with binary labels and exhibits moderate performance with multi-class dataset, NMI and ARI metrics suggest prominent increase in clustering quality regardless of target number of clusters.

2 Related Work

Unsupervised learning for text clustering has been extensively studied in feature representation (Mitra et al., 2002) and grouping methods (MacQueen et al., 1967; Von Luxburg, 2007). Classic text clustering approaches map text into a bag-of-words-based feature space and group text into clusters using methods like K-means (MacQueen et al., 1967) and Fuzzy C-means (Bezdek et al., 1984). These fast and widely-used approaches tend to be ineffective when inputs are sparse, high-dimensional, and semantically diverse (Yang et al., 2022). To explore better text clustering approaches, researchers have applied alternative feature representations like TF-IDF (Ramos et al., 2003) and word embedding (Wang et al., 2016), developed generative models for text grouping like the Latent Dirichlet allocation (LDA) model (Blei et al., 2003) and Gibbs Sampling Dirichlet Multinomial Mixture (GSDMM) model (Yin and Wang, 2014), and proposed algorithms that simultaneously learned feature representation and grouping, such as Deep Embedding Clustering (DEC) (Xie et al., 2016) and Deep Clustering Network (DCN) (Yang et al., 2017). In addition, Thompson and Mimno find that token level clustering with representation from pre-trained contextualized language models produces comparable performance to LDA topic models, with semantically richer representation of text information. Though the afore-mentioned approaches provide more handle distance matrixes by avoiding high-dimensional and sparse feature space, most of them fail to capture sequence and contextual information (Guan et al., 2020). For instance, they cannot distinguish the word love in sentences "You love her." and "You betrayed her love."

Deep learning-based text representation models have been used in recent studies to take both text sequence and contextual information into consideration (Guan et al., 2020). Compared with self-trained language models, state-of-the-art (SOTA) pre-trained models (i.e., BERT, GPT-3, T-5, etc.) are more generalized to various tasks and thus have strengthened the literature base in text clustering. A number of empirical studies have proved that SOTA pre-trained models are suitable for feature representation and that inputting features transformed by SOTA pre-trained models to clustering algorithm outperforms many existing clustering approaches (Guan et al., 2020; Subakti et al., 2022; Huang et al., 2020). For example, Guan et al. (2020) demonstrated that the clustering accuracy of pre-trained BERT with K-means was higher than generative models such as LDA and GSDMM. Huang et al. (2020) showed that a fine-tuned BERT with clustering loss (Kullback-Leibler (KL) divergence) could improve the clustering purity by 1-3% compared to pre-trained BERT. Subakti et al. (2022) applied different clustering algorithms and showed that BERT-based representation outperforms the TFIDF-based representation method in most scenarios. However, to this day, the potential of SOTA pre-trained models in text feature representation might be underestimated as the natural language processing techniques are still under development. Advanced pre-trained models such as RoBERTa proved to have a higher clustering purity than BERT (Aharoni and Goldberg, 2020).

New learning techniques such as few-shot learning methods have already been proposed to tackle many text tasks (Yan et al., 2018). One may argue that having few-shot learning in the landscape, the need for further discussions on unsupervised text clustering as a solution to labeling task is limited. Yet, empirical studies find simple clustering algorithms like K-means can produce comparable performance to state-of-the-art few-shot learning model and suggest potential pitfalls in current few-shot learning design (Masud Ziko et al., 2021). Also, one limitation of few-shot learning trained with a small amount of data is that the performance is sensitive to domain shift between train and test sets (Yao, 2021). Hence, we reckon text clustering with large pre-trained models is still worth investigating.

To fully tap the potential of SOTA pre-trained models on text representation for clustering, this study used classic clustering methods as baseline models and compared their performance with pre-trained text representation models based on various evaluation metrics. As mentioned in the introduction, to avoid the domain limitation in mannual labeling and the mislabeling problem, we will focus our methods on the advanced pre-trained models such as ELECTRA and DeBERTa.

3 Methodology

3.1 Data

We use the downsampled AG news and Yahoo! Answers datasets and the full R2 dataset to eval-

uate our models. The AG News dataset consists of four categories: **World**, **Sports**, **Business** and **Sci/Tech**, and for each category, there are 1000 training samples and 400 testing samples. We use both the title and the article as our input. The Yahoo! Answers dataset contains questions along with answers. There are ten categories: **Society** & **Culture**, **Science** & **Mathematics**, **Health**, **Education** & **Reference**, **Sports**, **Business** & **Finance**, **Entertainment** & **Music**, **Family** & **Relationships**, **Computer** & **Internet**, and **Politics** & **Government**, and each class contains 400 training samples and 100 testing samples. Since most of the questions include similar question words instead of useful information, we only use the best answer for each question as the input. The categories in R2 dataset are **earn** and **acq**, with class **earn** having 2877 training samples and 1087 testing samples, class **acq** having 1650 training samples and 719 testing samples.

3.2 Models

This project pipelines the modelling process into two sub tasks: 1) finding a proper text representation and 2) applying the clustering algorithms.

First of all, we establish some popular text clustering models as our baseline. Our project focuses on the following 3 clustering methods: K-means, Fuzzy C-Means (FCM) and Latent Dirichlet Allocation (LDA). These baseline models will use data representation from the TF-IDF method. The number of cluster centers are set to the number of categories in the data set.

Second, we retrieve data representation from some advanced language models and combine the text representation with the aforementioned 3 clustering algorithms. While BERT is a commonly discussed pre-trained language model in the field of natural language processing, we wish to visit more advanced models and examine how they will improve the text clustering performance. To the best of our knowledge, DeBERTa and ELECTRA have not been applied to the text clustering task in previous research papers. Hence, this paper focuses on analyzing the performance of these 2 pre-trained models, both of which have implementation ready for use in Hugging Face. Text representations from these models will replace the TF-IDF as inputs for the afore-mentioned clustering algorithms.

The last hidden layer of both pre-trained models and fine-tuned models are extracted for text representation. For the pre-trained models, we directly download weights from the website. For the fine-tuned models, we train the pre-trained models for 3 epochs in Hugging Face. We use 80% of the data as training data, set adjusted rand index (ARI) as the objective, and search the best learning rate from 1e-5 to 5e-5 with 3 trials. Considering the computational complexity, the maximum context length is set to 64.

3.2.1 Clustering Algorithms

K-Means K-means is one of the most popular unsupervised algorithms(MacQueen et al., 1967). The K denotes the number of centroids, the center of the clusters. The K-means algorithm keeps allocating all data points to the nearest cluster and updates the centroids by averaging over the data points within each cluster.

Fuzzy C-Means Fuzzy C-Means differs from the K-means algorithm as data points in Fuzzy C-Means do not belong to a single cluster and are not computed for only one cluster(Bezdek et al., 1984). Rather, all data points are associated to every cluster with different weighting. Hence, Fuzzy C-Means tends to run slower than K-means as Fuzzy C-Means needs to calculate distance weightings across the data points.

Latent Dirichlet Allocation Unlike the previous algorithms developed for the purpose of clustering, Latent Dirichlet Allocation is a generative statistical model that reveals latent clusters and assigns probabilities over the underlying clusters for all data points about their clustering membership (Blei et al., 2003).

3.2.2 Text Representation Algorithms

ELECTRA While BERT-style masked language modeling is a popular pre-training method, ELECTRA uses a different strategy. Instead of masking tokens and demanding more computation, ELECTRA opts for a more sample-efficient alternative that replaces some tokens with candidate tokens sampled from a small generator (Clark et al., 2019). ELECTRA delivers comparable performance to RoBERTa with smaller computation cost and hence is worth investigating in this project.

DeBERTa DeBERTa model extends the BERT model with disentangled attention and enhanced

mask decoder for more context and position information (He et al., 2020). Its latest version DeBERTaV3 further extends the performance by adopting an ELECTRA-style pretraining approach (He et al., 2021). In this project, we contrast the performance of DeBERTa model on text clustering task to basline TF-IDF and ELECTRA for a more comprehensive comparative analysis.

3.3 Evaluation Metrics

Text clustering task can not be simply evaluated by only generic metrics for the standard classification jobs like accuracy. Hence, we evaluate the performance using accuracy (ACC), normalized mutual information (NMI), and adjusted rand index (ARI). NMI compares the similarity between true labels and labels assigned by clustering algorithms for the same data while ARI focuses on the similarity between clusters. Higher scores in both metrics show that the algorithm has created finer clusters. For the following expressions, y is the ground-truth label and $\hat{y}_i$ is the cluster label.

ACC

$$\text{ACC} = \frac{\sum_i^n \delta(y_i, \hat{y}_i)}{n}$$

where the function $\delta(y_i, \hat{y}_i) = 1$ when $y_i = \hat{y}_i$ and 0 otherwise.

NMI

$$\text{NMI}(y_i, \hat{y}_i)) = \frac{I(y_i, \hat{y}_i))}{\sqrt{H(y_i)H(\hat{y}_i))}}$$

where I is the mutual information between $y_i, \hat{y}_i)$, and $H(y_i), H(\hat{y}_i))$ are the entropies of $y_i, \hat{y}_i)$.

ARI ARI is a similarity measure between two clusterings.

$$\text{RI} = \sum_{ij} \binom{n_{ij}}{2}$$

$$\max(\text{RI}) = \frac{1}{2}\Big[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\Big]$$

$$\text{Expected RI} = \Big[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\Big] / \binom{n}{2}$$

$$\text{ARI} = \frac{\text{RI} - \text{Expected RI}}{\max(\text{RI}) - \text{Expected RI}}$$

where n_{ij}, a_i, b_j are values from contingency table.

4 Results and Discussion

We summarize and present the experiment results evaluated by ACC, NMI, and ARI in the Appendix section for reference. Empirical study shows that the clustering performance is greatly improved compared to baseline across all 3 datasets with the text representation retrieved from fine-tuned large language models. While ELECTRA and DeBERTa outperforms baseline models by capturing both basic linguistic pattern and generalized information from the corpus, the untuned pre-trained setting does not always beat the baseline. Taking R2 dataset as an example, the NMI and ARI can be as low as 0.017 and 0.026 with `pre-trained ELECTRA+LDA` set-up. Yet, after fine-tuning the ELECTRA model, the NMI and ARI drastically increase to 0.888 and 0.943 respectively, indicating the versatility of large pre-trained models on text clustering task with a few adjustments.

Text clustering models also demonstrate different performances for different datasets. Among all 3 datasets, R2 dataset exhibits the highest clustering quality for all models. We attribute this observation to the fact that R2 dataset has only 2 classes to differentiate and text representations of R2 are intuitively more distant from each other. For pre-trained ELECTRA and DeBERTa which we reckon to be less representative, distance-sensitive clustering algorithm Fuzzy C-Means stands out with significant higher NMI and ARI. While the results for AG news and Yahoo! Answers are not comparable to R2 dataset, we still see significant improvements in evaluation metrics with ELECTRA and DeBERTa compared to baseline, confirming the effectiveness of these large pre-trained language models on text clustering task.

Though the evaluation metrics show small variations in value for ELECTRA and DeBERTa, we do not see a significant difference of clustering performance in these 2 models. For future work, one major question to answer is how we can better deploy text clustering models for multiclass datasets with cross domain contexts.

5 Collaboration Statement

All team members participated in background research, project design and report write-up.

6 Ethical Statement

Despite the wide applications of text clustering, there are several ethical considerations need to be

well addressed. First, the machine learning models such as DeBERTa and ELECTRA may bring bias due to the data they are trained on. In detail, the data used for text clustering are mostly human languages which reflect the stereotypical human biases, i.e. bias in gender/race. Selecting data from multi-sources and experimenting on two or more alternatives may alleviate the issue. Second, the data that are used for text clustering problems usually come from social media, which are not technically private but sensitive in some ways as users can inadvertently share the information that compromises the personal security. Researchers need to make an agreement with the users about the collection and usage of the data, and the potential risks from future research. Although it is hard to know the disparate impact at the very beginning, researchers can consider these ethical issues throughout the research to reduce any potential impact.

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. pages 7747–7763.

James C Bezdek, Robert Ehrlich, and William Full. 1984. Fcm: The fuzzy c-means clustering algorithm. *Computers & geosciences* 10(2-3):191–203.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3(Jan):993–1022.

Christos Bouras and Vassilis Tsogkas. 2017. Improving news articles recommendations via user clustering. *International Journal of Machine Learning and Cybernetics* 8(1):223–237.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. pages 2147–2157.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. 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)*. pages 4171–4186.

Renchu Guan, Hao Zhang, Yanchun Liang, Fausto Giunchiglia, Lan Huang, and Xiaoyue Feng. 2020. Deep feature-based text clustering and its explanation. *IEEE Transactions on Knowledge and Data Engineering* .

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543* .

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Michael Edward Houle and Nizar Grira. 2007. A correlation-based model for unsupervised feature selection. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. pages 897–900.

Shaohan Huang, Furu Wei, Lei Cui, Xingxing Zhang, and Ming Zhou. 2020. Unsupervised fine-tuning for text clustering. In *Proceedings of the 28th International Conference on Computational Linguistics*. pages 5530–5534.

Yong Huang, WeiJing Huang, XiaoLin Xiang, and JinJiang Yan. 2021. An empirical study of personalized advertising recommendation based on dbscan clustering of sina weibo user-generated content. *Procedia Computer Science* 183:303–310.

Alexander A Kharlamov, Andrey V Orekhov, Svetlana S Bodrunova, and Nikolay S Lyudkevich. 2019. Social network sentiment analysis and message clustering. In *International Conference on Internet Science*. Springer, pages 18–31.

Chung-Hong Lee and Hsin-Chang Yang. 2009. Construction of supervised and unsupervised learning systems for multilingual text categorization. *Expert Systems with Applications* 36(2):2400–2410.

James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Oakland, CA, USA, volume 1, pages 281–297.

Imtiaz Masud Ziko, Malik Boudiaf, Jose Dolz, Eric Granger, and Ismail Ben Ayed. 2021. Transductive few-shot learning: Clustering is all you need? *arXiv e-prints* pages arXiv–2106.

Pabitra Mitra, CA Murthy, and Sankar K. Pal. 2002. Unsupervised feature selection using feature similarity. *IEEE transactions on pattern analysis and machine intelligence* 24(3):301–312.

Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*. Citeseer, volume 242, pages 29–48.

Minoru Sasaki and Hiroyuki Shinnou. 2005. Spam detection using text clustering. In *2005 International Conference on Cyberworlds (CW'05)*. IEEE, pages 4–pp.

Eyal Shnarch, Ariel Gera, Alon Halfon, Lena Dankin, Leshem Choshen, Ranit Aharonov, and Noam Slonim. 2022. Cluster & tune: Boost cold start performance in text classification. *arXiv preprint arXiv:2203.10581* .

Dominik Stammbach and Elliott Ash. 2021. Docscan: Unsupervised text classification via learning from neighbors. *Center for Law & Economics Working Paper Series* 2021(08).

Alvin Subakti, Hendri Murfi, and Nora Hariadi. 2022. The performance of bert as data representation of text clustering. *Journal of big Data* 9(1):1–21.

Xiaohui Tao, Patrick Delaney, and Yuefeng Li. ???? Text categorisation on semantic analysis for document categorisation using a world knowledge ontology .

Laure Thompson and David Mimno. 2020. Topic modeling with contextualized word representation clusters. *arXiv preprint arXiv:2010.12626* .

Ulrike Von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and computing* 17(4):395–416.

Peng Wang, Bo Xu, Jiaming Xu, Guanhua Tian, Cheng-Lin Liu, and Hongwei Hao. 2016. Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing* 174:806–814.

Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*. PMLR, pages 478–487.

Leiming Yan, Yuhui Zheng, and Jie Cao. 2018. Few-shot learning for short text classification. *Multimedia Tools and Applications* 77(22):29799–29810.

Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. 2017. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *international conference on machine learning*. PMLR, pages 3861–3870.

Shuiqiao Yang, Guangyan Huang, Bahadorreza Ofoghi, and John Yearwood. 2022. Short text similarity measurement using context-aware weighted biterms. *Concurrency and Computation: Practice and Experience* 34(8):e5765.

Yiming Yang and Jan O Pedersen. 1997. A comparative study on feature selection in text categorization. In *Icml*. Nashville, TN, USA, volume 97, page 35.

Fupin Yao. 2021. Cross-domain few-shot learning with unlabelled data. *arXiv preprint arXiv:2101.07899* .

Jianhua Yin and Jianyong Wang. 2014. A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. pages 233–242.

A Appendix

Model	ACC	NMI	ARI
R2 dataset			
TF-IDF + K-means	0.873	0.555	0.557
TF-IDF + FCM	0.874	0.556	0.558
TF-IDF + LDA	0.924	0.656	0.719
pre-trained DeBERTa + K-means	0.886	0.487	0.597
pre-trained DeBERTa + FCM	0.919	0.624	0.703
pre-trained DeBERTa + LDA	0.914	0.598	0.686
fine-tuned DeBERTa + K-means	0.984	0.878	0.937
fine-tuned DeBERTa + FCM	0.984	0.878	0.937
fine-tuned DeBERTa + LDA	0.984	0.882	0.939
pre-trained ELECTRA + K-means	0.62	0.018	0.027
pre-trained ELECTRA + FCM	0.885	0.479	0.592
pre-trained ELECTRA + LDA	0.619	0.017	0.026
fine-tuned ELECTRA + K-means	0.986	0.888	0.943
fine-tuned ELECTRA + FCM	0.986	0.888	0.943
fine-tuned ELECTRA + LDA	0.986	0.892	0.945
AG News dataset			
TF-IDF + K-means	–	0.055	0.025
TF-IDF + FCM	–	0.109	0.079
TF-IDF + LDA	–	0.218	0.199
pre-trained DeBERTa + K-means	–	0.414	0.385
pre-trained DeBERTa + FCM	–	0.255	0.196
pre-trained DeBERTa + LDA	–	0.47	0.469
fine-tuned DeBERTa + K-means	–	0.708	0.751
fine-tuned DeBERTa + FCM	–	0.708	0.751
fine-tuned DeBERTa + LDA	–	0.706	0.748
pre-trained ELECTRA + K-means	–	0.046	0.037
pre-trained ELECTRA + FCM	–	0.016	0.015
pre-trained ELECTRA + LDA	–	0.04	0.039
fine-tuned ELECTRA + K-means	–	0.699	0.737
fine-tuned ELECTRA + FCM	–	0.696	0.737
fine-tuned ELECTRA + LDA	–	0.704	0.741
Yahoo! Answes dataset			
TF-IDF + K-means	–	0.066	0.018
TF-IDF + FCM	–	0.041	0.014
TF-IDF + LDA	–	0.033	0.007
pre-trained DeBERTa + K-means	–	0.122	0.052
pre-trained DeBERTa + FCM	–	0.018	0.008
pre-trained DeBERTa + LDA	–	0.113	0.046
fine-tuned DeBERTa + K-means	–	0.378	0.295
fine-tuned DeBERTa + FCM	–	0.196	0.109
fine-tuned DeBERTa + LDA	–	0.347	0.284
pre-trained ELECTRA + K-means	–	0.044	0.01
pre-trained ELECTRA + FCM	–	0.031	0.01
pre-trained ELECTRA + LDA	–	0.041	0.014
fine-tuned ELECTRA + K-means	–	0.333	0.231
fine-tuned ELECTRA + FCM	–	0.181	0.089
fine-tuned ELECTRA + LDA	–	0.323	0.238

Table 1: Experiment results. For R2 dataset, fine-tuned models outperform pre-trained models while both types of models beat the baseline. From the high accuracy on R2, we can conclude that with fine-tuned models, all 3 clustering algorithm can categorize most data correctly. For AG News dataset and Yahoo! Answers dataset, since both of them contain more than 2 categories while clustering algorithms can only assign data with the cluster number instead of the right label, accuracy cannot fairly evaluate the performance of algorithms. Still, NMI and ARI scores show that fine-tuned models outperform in this task for both AG News dataset and Yahoo! Answers dataset. Also, with more categories, it becomes harder for clustering algorithms to perform correct categorization.