Text Classification with Named-Entity Recognition and AutoPhrase

Siyu Deng [a] and **Yang Li**[b] and **Rachel Ung**[c]
University of California, San Diego,
[a] sid015@ucsd.edu, [b]yang@ucsd.edu, [c]rung@ucsd.edu

Abstract

Text Classification (TC) and Named-Entity Recognition (NER) are two fundamental tasks for many Natural Language Processing (NLP) applications, which involve understanding, extracting information, and categorizing the text. In order to achieve these goals, we utilized AutoPhrase[2] and a pre-trained language NER model[3] to extract quality phrases. Using these as part of our features, we are able to achieve very high performance for a five-class and a twenty-class text classification dataset. Our project[1] will follow a similar setting as previous works with train, validation, and test datasets and comparing the results across different methods.

Keywords: Natural Language Processing, Named-Entity Recognition, Information Extraction, Text Classification

1 Introduction

Text classification is an important NLP task, which can be understood as a given set of texts and labels. We want to create a classifier that can classify these given inputs in addition to other texts. Text classification tasks mainly involve understanding the text and extracting high quality phrases for model training. For example, if a text has "government" or "minister" as a frequent phrase or word, it is more likely to belong to 'Politics'. As such, it is important for us to extract quality phrases and make sure they represent these documents well.

Named-Entity Recognition and Phrase-Mining are some methods to extract quality phrases. For NER, the task mainly involves locating and classifying the word in the text to entities that are predefined categories and domains, e.g. "“Bill Gates” – Person; “Sandag” – Organization. A well-trained NER model is optimal to extract entities. One of our hypotheses is that these entities are representative of the text. To further improve our classifier, we would like to incorporate a phrase-mining task in our project. AutoPhrase[2] is a method that can extract quality phrases from a corpus with a ranking, without the need for any human annotations. As we have learned from experimenting from last quarter, it is very efficient and powerful compared to many existing methods. By extracting such quality phrases, we believe we could build a stronger classifier for text classification.

We are motivated by our interest in exploring language related NER models. We believe this project is an excellent opportunity for both exploring language models, Named-Entity Recognition tasks, and Text Classification tasks. By utilizing the Phrase-Mining and NER tools, we are able to achieve very high performance on both BBC and 20 Newsgroups dataset. In parallel, we conduct an analysis and compare the performance between each of the methods.

2 EDA

We began this project by investigating NER problems and found many existing methods have been trained on the CoNNL2003 dataset. This dataset contains 20,744 English sentences and four major types of entities: PER (person), LOC (location), ORG (organizations), and MISC (entities not included in the previous three).

After experimenting with training a Named-Entity Recognition model, we found that it was a difficult and time-consuming task. Considering the quarter-long time constraint and our current skill set, we decided to direct our focus to Text Classification.

[1] Github: tinyurl.com/55mnsfea

For text classification, we have used two datasets: a BBC News dataset [2] and a 20 Newsgroups dataset[3], which are commonly used for text classification tasks.

- BBC News dataset: includes 2,225 documents and spans 2004-2005; composed of five categories (entertainment, technology, politics, business, and sports.

FEATURE	DESCRIPTION
'text'	a BBC article corpus
'target'	the Text Classification
'summary'	summary of corpus

Table 1: Features of the BBC News Dataset

- 20 Newsgroups dataset: includes 18,000 Newsgroups posts; composed of 20 categories (Computer, Science, Politics, Religion, etc.)

FEATURE	DESCRIPTION
'post'	a post-formatted corpus
'target'	the Text Classification
'target names'	name of the document target

Table 2: Features of the 20 Newsgroups Dataset

As we can see from Figure 1, for each category, there are about 400 - 500 news reports; the dataset is fairly balanced between the categories.

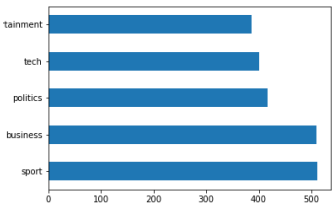

Figure 1: Documents Counts for Each Category from BBC news

Based on the results from Figure 2, the range of the summary text is about 10 - 250 tokens, meaning it can be considered normally distributed. For the shorter length entries, they may face more issues when performing the Text Classification. We will compare the results between the different lengths.

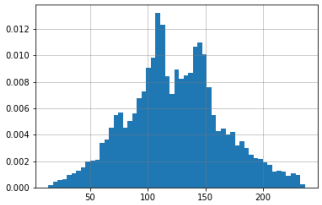

Figure 2: Summary Text's Sentence Lengths from BBC News

Regarding Figure 3, the word clouds indicate that there are some words that appear more frequently than others in each category; this may play a significant role in how the documents are classified.

Figure 3: Word Clouds for each News Category from BBC News

3 Feature Engineering

For all feature engineering, we are using the traditional Bag-of-Words and Term Frequency–Inverse Document Frequency (TF-IDF) representations to encode the entities and quality phrases into vectors. The major difference is the vocabulary pool we are using for the different feature engineering.

3.1 Entity-based Feature

BERT (Bidirectional Encoder Representations from Transformers) is a general-purpose language model trained on the large dataset. This pre-trained model can be fine-tuned and used for different tasks such as sentiment analysis, question answering systems, sentence classification, and Named-Entity Recognition. Named-Entity Recognition is the process of extracting noun entity from text data and classifying them

[2]BBC News Article Dataset: `https://www.kaggle.com/pariza/bbc-news-summary`

[3]20 Newsgroups Dataset: `http://qwone.com/~jason/20Newsgroups/`

into predefined categories e.g. person, location, organization and others. Hence, we can use a BERT-based Named-Entity Recognition model, fine-tuned on the CoNLL 2003 dataset, to extract noun entities in the BBC News data set and 20 News group datasets.

For our experiment, we have used the BERT-based uncased model as a baseline trained by the HuggingFace library with 110M parameters, 12 layers, 768-hidden, and 12-heads. For fine-tuning, we used the suggested parameters of $max\ seq\ length = 128$, $training\ epoch = 3$, and $warmup\ proportion = 0.1$. Then, we created the dataframe for BBC News summary data and used the model to predict the entity by each sentence of the document. We followed the same procedure for the 20 News Dataset.

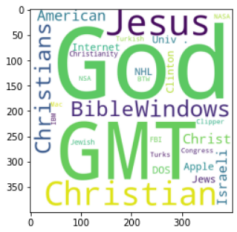

Figure 4: Entity Cloud (20 Newsgroups)

3.2 AutoPhrase-based Feature

AutoPhrase[2] is a method that builds upon SegPhrase [1], that is a completely automated technique that can extract quality phrases from massive text. Unlike SegPhrase, which requires human labeling, AutoPhrase utilizes Wikipedia, or similar existing knowledge bases, and raw corpora as input.

AutoPhrase has two major modules: Robust Positive-Only Distant Training and POS-Guided Phrasal Segmentation. The first module trains the model and determines the quality score for phrases; the second module determines which tokens should be combined together and constitute a phrase. AutoPhrase first estimates the quality score from frequent n-gram phrases. With these results, it then utilizes the segmentation module to revise the segmentation. Rather than using the n-gram based phrases, AutoPhrase estimates the final quality score again based on the segmentation results.

Since AutoPhrase is applicable to any domains and languages, we are able to utilize this method on both of our datasets to extract quality phrases. From Table 3 below, these are high quality phrases extracted from the 20 Newsgroups dataset, with many phrases especially for the sports and politics groups. We believe these are useful features to include in our experiment.

p-value	Phrases
0.9888640523	george bush
0.9873576373	red sox
0.9863977181	attorney general
0.9858852675	gulf war
0.9856553314	silicon graphics
0.9853577556	vice president
0.9849079262	united nations
0.9846998338	soviet union
0.9845569952	north america
0.9842812319	south africa

Table 3: Top 10 Quality Phrases by AutoPhrase (20 Newsgroups)

4 Model & Experiment

4.1 Logistic Regression

Logistic Regression is a binary classifier that is widely adopted for many research projects and real-world applications. As such, we decided to incorporate this model in our experiment as well. This model is optimized by minimizing the Logistic Loss (Equation 1).

$$L(y_i, \widehat{y}_i) = -(y_i log(\widehat{y}_i) + (1 - y_i) log(1 - \widehat{y}_i)) \quad (1)$$

Since Logistic Regression is a binary classifier, we have used a One-Verses-Rest strategy in this multi-class classification task. This means training a binary classifier for each class (e.g. Does this text belong to the Sports group?)

4.2 Support Vector Machine

A Support Vector Machine (SVM) is a supervised model intended for solving classification problems. The SVM algorithm creates a line or a hyper-plane, which separates the data into classes. This model is optimized by minimizing the Hinge Loss (Equation 2)

$$\lambda |w|_2^2 + \sum_{i=1}^{n} max(0, 1 - y_i(w^T x_i - b) \quad (2)$$

4.3 BERT

The architecture of BERT's transfer learning is made up by a fully-connected layer, a drop-out layer, a Rectified Linear Unit (ReLU) activation layer, a second fully-connected layer, and a soft-max activation layer. For the optimizer, we used AdamW, an improved version of Adam, and opted to use the negative log-likelihood loss, which is well-suited for multiple-class classification. For training, we used a learning rate

of $1e^{-4}$ for 40 epochs. Due to GPU resources, we were only able to perform training and evaluation on the BBC News dataset.[4]

4.4 Experiment

For both datasets, we have adopted a train/validation/test split. For the 20 News group dataset, we used the sklearn library and then applied a 50% / 50% on the test section. For the BBC news dataset, we adopted the 60% / 20% / 20% split.

We tested the combinations of two major models (LOG and SVM) over three vocabulary lists. We used Bag-of-Word models as our main baseline, as we can see from the results below. Our model combining Unigrams, Entities, and AutoPhrase features performs the best across all models.

5 Results

For the models constructed, the vectors are generated using Bag-of-Words or Tf-Idf and vary by their vocabulary lists.

There are four main vocabulary lists:

- Uni-gram Vocabulary List (UNI)
- Entity Vocabulary List (ET)
- AutoPhrase Vocabulary List (AP)
- All Vectors Vocabulary List (ALL)

Three Models:

- Logistic Regression (LOG)
- Support Vector Machine (SVM)
- BERT (BR)

Model	Weighted F1	Accuracy
LOG + UNI(BOW)	0.9527	0.9528
SVM + UNI(BOW)	0.9485	0.9483
SVM + ET(TF-IDF)	0.9529	0.9528
SVM + AP(TF-IDF)	0.9462	0.9461
SVM + ALL(TF-IDF)	**0.9639**	**0.9640**

Table 4: BBC News Dataset Validation Result

Model	Weighted F1	Accuracy
LOG + UNI(BOW)	0.7751	0.7759
SVM + UNI(BOW)	0.7629	0.7589
SVM + ET(TF-IDF)	0.8259	0.8282
SVM + AP(TF-IDF)	0.8105	0.8125
SVM + ALL(TF-IDF)	**0.8466**	**0.8483**

Table 5: 20 Newsgroups Dataset Validation Result

BERT Classification on BBC News Data

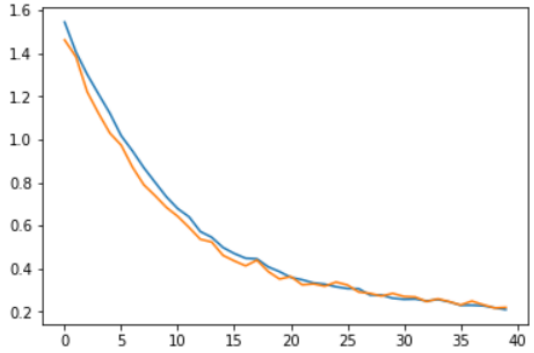

Figure 5: BERT on BBC News Train-Validation Loss Curve

	precision	recall	f1-score	support
0	0.83	0.87	0.85	77
1	0.86	0.95	0.90	58
2	0.91	0.85	0.88	62
3	0.99	0.94	0.96	77
4	0.90	0.87	0.88	60
accuracy			0.90	334
macro avg	0.90	0.89	0.89	334
weighted avg	0.90	0.90	0.90	334

Figure 6: BERT on BBC News Classification Report

6 Conclusion

The BERT classification on the five-class BBC News dataset does not outperform any of our implemented models. From our results table, we observed that our models have F1-Score and Accuracy performances at around 0.95, indicating they are high-performing classifiers. The best of them is the SVM+ALL(TF-IDF) classifier, or the Support Vector Machine with the All Vector Vocabulary List and Tf-Idf Representations, which uses the vocabulary from both NER results and AutoPhrase results. Because the quality phrases between different domains are likely to differ, we expect these results to be optimal features for our predictors.

For the 20 News Group dataset, the SVM+ALL(TF-IDF) classifer also outperformed the other models, with the F1-Score and Accuracy being 0.84. Considering the classes are huge (i.e. 20 classes), these results verify our model is high-performing. Applying our best model on the five-class BBC News dataset, we attained a F1-Score at 0.9525, and Accuracy at 0.9528; while for the 20 News Group, we yielded a F1-Score at 0.8463 and Accuracy at 0.8478.

In this project, we utilized AutoPhrase and pre-trained BERT NER model for text classification. The results are pretty powerful in terms of accuracy and F1 score. We do think Entity and Quality Phrases are very powerful features to use for text classification task. However, if we include the uni-gram feature as well we would achieve much better performance.

7 Acknowledgments

We would like to give special thanks to our mentor Professor Jingbo Shang, who guided us our project and provided constructive suggestions alone the way. We would also like to give special thanks to Professor Aaron Fraenkel and our teaching assistants, who have given us meaningful lectures about structuring our data science project and providing suggestions from our code to our presentation.

References

[1] Jialu Liu*, Jingbo Shang*, Chi Wang, Xiang Ren and Jiawei Han, "Mining Quality Phrases from Massive Text Corpora", Proc. of 2015 ACM SIGMOD Int. Conf. on Management of Data (SIGMOD'15), Melbourne, Australia, May 2015. (* equally contributed)

[2] Jingbo Shang, Jialu Liu, Meng Jiang, Xiang Ren, Clare R Voss, Jiawei Han, "Automated Phrase Mining from Massive Text Corpora", accepted by IEEE Transactions on Knowledge and Data Engineering, Feb. 2018.

[3] Thomas Wolf and Lysandre Debut and Victor Sanh and Julien Chaumond and Clement Delangue and Anthony Moi and Pierric Cistac and Tim Rault and Rémi Louf and Morgan Funtowicz and Joe Davison and Sam Shleifer and Patrick von Platen and Clara Ma and Yacine Jernite and Julien Plu and Canwen Xu and Teven Le Scao and Sylvain Gugger and Mariama Drame and Quentin Lhoest and Alexander M. Rush, "Transformers: State-of-the-Art Natural Language Processing", in Proc. of the 2020 Conference on Empirical Methods in Natural Language (EMNLP' 20).

[4] "Transfer Learning for NLP: Fine-Tuning BERT for Text Classification". https://www.analyticsvidhya.com/blog/2020/07/transfer-learning-for-nlp-fine-tuning-bert-for-text-classification/