Optimal Model Input for Newspaper Topic Classification

Carmen Easterwood
W266 Natural Language Processing with Deep Learning
Summer 2018

Abstract

In this paper, I develop a topic classification model for news articles. I test multiple different model inputs, but find that feeding the full article text into the model still generates the best results. The highest performing model (so far) is a logistic regression model with accuracy of 77%, and F1 scores to be calculated for the final paper.

1 Introduction

Classifying news articles by topic is an important form of content analysis. It lets us know what journalists are writing about, whether that is changing over time, and the relative important of different topics. Tagging articles by topic also makes article searches more efficient for news consumers and researchers. However, manual topic-coding is quite time-consuming, so automating this process can save time and money for news organizations. This paper will develop a supervised topic classification model using New York Times news articles.

2 Background

There is a large literature looking at various aspects of document classification, but only a few have focused on evaluating newspaper articles. Martin and Johnson (2015) used a dataset from the San Jose Mercury News and found that using a nouns-only or lemmatized version of the articles improved topic classification. However, theirs was an unsupervised model, so they evaluated their model using techniques that are not applicable in a supervised setting.

Meanwhile, Wermter and Hung (2002) created a semi-supervised self-organizing memory (SOM) model for topic generation of Reuters news articles. They used WordNet relationships to assist their model and thus limited their model input to only the nouns and verbs found in WordNet. This allowed them to achieve accuracies over 95%, but generalizability may be limited due to this use of WordNet.

Other recent papers have looked at supervised topic classification for non-news documents. Karan et al (2016) looks at classification of Croatian political texts) using word2vec, logistic regression, Gaussian NB, and gradient-boosted trees, as well as a couple of postprocessing rules, to reach an F1 score of 77% for major topics. Glava et al (2017) looks at SVM and CNN models for multiple languages, and for monolingual English models finds that CNNs perform best with an accuracy of 57%. In this paper, I will test whether some of the ideas and techniques used for classification of non-news documents can be successfully extended to the news article setting.

3 Methods

3.1 Dataset

I used the New York Times Annotated Corpus, which contains 1.8 million articles from 1987-2007, labeled with topics, subtopics, and newspaper sections. Due to computing power limitations, I have selected a random sample of 10,000 articles to parse and use as data for the model. The articles are randomly split so that approximately 75% are training data, 5% are development data, and 20% test data.

3.2 Model Input

Based on my review of the literature, I tested five different model inputs, some of which are specific to the newspaper context. Table 1 shows some key statistics for each type of model input for my random sample of 10,000 articles.

1. Full text of article

Model Input	Training Words			Cleaned Label Vocab (Unique Words)		
	Count	% Full Text	Avg. per Article	Count	% Full Text	Avg. per Article
Full Text	49.7M	100	672	87,541	100	1,172
Lead Paragraph	7.5M	15	104	40,309	48	561
Headlines	0.6M	1	8	11,150	15	154
Nouns	13.3M	27	180	64,998	80	1,000
Lemmas	49.7M	100	672	75,648	94	1,000

Table 1: Model Inputs

Desk	General Descriptor	Online Sections	Taxonomic Classifier
Foreign Desk	• Immigration and Refugees • Jews • Music • Religion and Churches	• World	• Top/News/World/Europe • Top/News/World/Countries and Territories/Austria • Top/Features/Arts/Music • Etc. (8 others)
Book Review Desk	• Books and Literature	• Arts • Books	• Top/Feature/Arts • Top/Features/Books • Top/Features/Books/Book Reviews
Classified		• Paid Death Notices	• Top/Classifieds/Paid Death Notices

Table 2: Possible Model Outputs

2. Nouns-only version of the full text (Martin and Johnson, 2015)

3. Lemmatized version of the full text (Martin and Johnson, 2015)

4. Article's headline (Wermter and Hung, 2002)

5. Article's lead paragraph, which is supposed to hook the reader, and in a news context often summarizes important details of the story (Bloch, 2016)

3.3 Model Output

The NYT dataset has multiple ways of classifying articles. Table 2 shows examples of the four different ways that six articles were classified. In my models I use the `desk` column as my model output variable because:

1. It has the lowest percentage of null values

2. It never assigns multiple descriptions to the same article

However, `desk` still requires some clean-up before use due to misspellings and possible changes to the desk names over time. Table 3 shows some examples of how I cleaned the `desk` variable so the model would have a standardized set of output labels for the articles.

After cleaning, roughly 95% of articles fall into the top 20 categories, and then there are a large number of very small categories with just a couple of articles. I therefore created an `<OTHER>` category as a catchall for these tiny categories,

Original Label Name	Articles
Book Review Desk	1,750
book review	1
Book Review Post	4
Business Desk	8
Business World Magazine	6,078
Business/Financial Desk	1
Business/Financial Desk;	1
Business/Financial desk	5
Business/FinancialDesk	2
Business\Financial Desk	6
E-Business	17
E-Commerce	11,276
Financial Desk	16
Financial Desk;	3
Money & Business/Financial Desk	1
Money and Busines/Financial Desk	939
Money and Business/Financial Desk	1
Moneyand Business/Financial Desk	12
Small Business	82
SundayBusiness	3
The Business of Green	122
Automobiles	28
Cars	

Table 3: Label Cleaning Examples

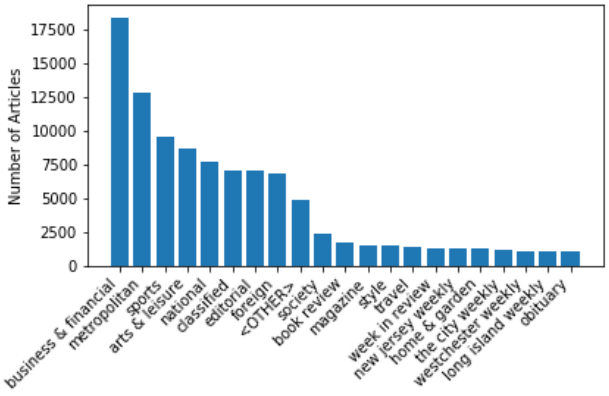

Figure 1: Frequency of Top 20 Labels + `<OTHER>`

which individually don't have enough data for a model to learn very well. Figure 1 shows the frequency distribution of the top 20 category labels, plus `<OTHER>`.

3.4 Models

For each model input, I tested three models:

1. Multinomial naïve Bayes

2. Multi-class logistic regression (one vs. rest)

3. Convolutional neural network

For the naive Bayes and logistic regression models, I tested multiple parameter values on the dev data, and chose the parameter values that resulted in the highest accuracy. Figures 2 and 3

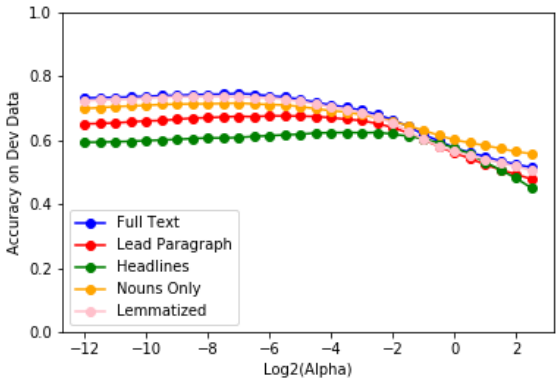

Figure 2: MNB Accuracy on Dev Data

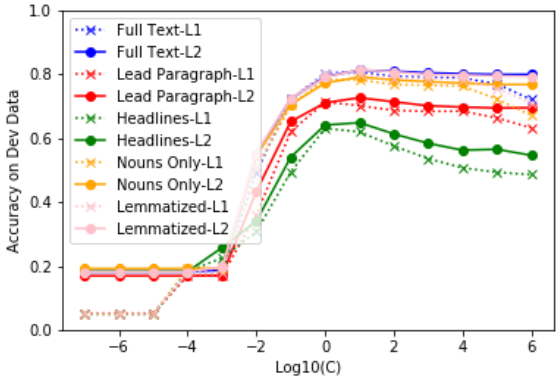

Figure 3: LR Accuracy on Dev Data

show how the models performed with different parameter values.

(Glavaš et al., 2017) – (Glavaš et al., 2017) – Glavaš et al. (2017) – Glavaš et al., 2017 – (2017) – Glavaš et al. – (2017) – (Glavaš et al., 2017; Wermter and Hung, 2002)

(Karan et al., 2016)

(Martin and Johnson, 2015)

(Pennington et al., 2014)

"Stuff ... "

4 Results & Discussion

Note: (Wermter and Hung, 2002) uses only top 8 topics.

5 Conclusion

No time for: SVM, gradient boosted trees

An interesting future project would be to run an unsupervised topic classification algorithm on this corpus (e.g. Martin and Johnson (2015) use the Latent Dirichlet Allocation algorithm) and compare those results to the supervised learning results.

Model Input	**Model Type**			**Best Model**
	MNB	**LR**	**CNN**	
Full Text	**0.729**	**0.815**	0.190	LR
Lead Paragraph	0.682	0.731	**0.248**	LR
Headlines	0.622	0.647	0.173	LR
Nouns	0.713	0.786	0.185	LR
Lemmas	0.727	0.811	0.189	LR
Best Input	Full Text	Full Text	Lead Para	**Full Text in LR**

Table 4: Model Accuracies on Test Data

Model Input	**Vocab Size**	**Padding Size**
Full Text	207,070	500
Lead Paragraph	99,210	139
Headlines	30,658	11
Nouns	165,531	378
Lemmas	193,694	500

Table 5:

References

Hannah Bloch. 2016. A good lead is everything – here's how to write one. *NPR.org*.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2017. Cross-lingual classification of topics on political texts. In *Proceedings of the Second Workshop on Natural Language Processing and Computational Social Science*, pages 42–46. Association for Computational Linguistics.

Mladen Karan, Jan Šnajder, Daniela Širinić, and Goran Glavaš. 2016. Analysis of policy agendas: Lessons learned from automatic topic classification of croatian political texts. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences and Humanities (LaTeCH)*, pages 12–21. Association for Computational Linguistics.

Fiona Martin and Mark Johnson. 2015. More efficient topic modelling through a noun only approach. In *Proceedings of Australasian Language Technology Association Workshop*, pages 111–115.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Stefan Wermter and Chihli Hung. 2002. Selforganizing classification on the reuters news corpus. In *Proceedings of the 19th international conference on computational linguistics - Volume 1*.