Deep Learning based Authorship Identification

Arth Talati ATALATI@SEAS.UPENN.EDU
Anirudh Sharma CA ANI22@SEAS.UPENN.EDU
Ranjani Narayanan RANJANIN@SEAS.UPENN.EDU

Abstract

In this project, we apply basic classification models and explore GRU, LSTM and Bi-LSTM at the sentence and article levels to identify the authors of a given piece of text. We deal with the pre-processing and feature vectorization of texts from the Reuters_50_50 C-50 data-set. The features from these vectors are extracted and word embeddings using GloVe are created. The model is developed to deal with larger pieces of textual information and analyse semantic and metaphorical uses of words with the goal of improving authorship identification tasks.

1. Introduction

This project deals with the extraction of information from Reuter_50_50's C-50 data-set of news-wire stories with over 10000 attributes for pattern recognition in texts. This data-set was chosen since it consists of 50 authors of texts labeled with at least one subtopic of the corporate/industrial class, in an attempt to alleviate the effects of "topic" in distinguishing among texts. The training and test batches each consist of 5000 texts, 100 per author, that are non-overlapping with each other.

Firstly, the quality of the dataset is inspected along with running a thorough analysis of the features to be used. Pre-processing of the dataset with TF-IDF implementation is conducted at the article level for every author in the training and testing datasets. Basic classification models are then trained so as to leverage the data-set features and a higher-level implementation (LSTM GRU). A comparative analysis of the results of each of these models is conducted, using different analytical metrics, and future scope for implementation of the project is suggested.

University of Pennsylvania, CIS 419/519 Course Project.

2. Related Work

Currently a lot of research in NLP is focused on Authorship identification. [1] provides deep learning methods for authorship identification for Reuters 50-50 and Gutenberg Datasets, most of our work has been inspired from [1]. [2] deals with utilizing Recurrent Neural networks for News authorship identification. Similar approaches for authorship identification but utilizing stacked de-noising auto-encoders for feature extraction for SVM classification has been implemented in [3]. Authorship verification methods have been transferred to authorship identification in [4], which is one of the most effective methods utilizing documents from external sources. [5] provides a brief overview of a multi-headed recurrent neural network for authorship identification. A basic tool to diagnose utilizing LSTM neural networks has been implemented in [6]. The references above have helped us implement deep learning models for authorship identification.

3. Data Pre-Processing

Depending on the type of the dataset, we used some of the following pre-processing functions on our text data.

3.1. Feature set

Varieties in input capitalization yield various kinds of output or no output. This may occur if the dataset contains blended case events of words and there is little evidence for the neural-network to adequately learn the weights for the less common versions. Given that news articles may sometimes consist of casing errors, lower-casing has been implemented here to improve the efficiency of performance tasks. Instances of stop words in English viz. "a", "the", "is", "are" and so forth are eliminated since they are low information words. We attempt to diminish the number of features that help keep our models within manageable sizes by their removal. In order to ensure high information gain, we have only retained those words that appear in at least 5 documents and ignored those words that frequent in more than 50% of the documents.

Table 1. Stylistic Features in the dataset

FEATURE	FEATURE
FULL-STOPS	APOSTROPHES
COMMA	LINE BREAKS
QUESTION MARKS	WHITE SPACES
EXCLAMATION MARKS	ELLIPSIS
AMPERSANDS	LEFT CURLY BRACKETS
PERCENTAGE SIGN	RIGHT CURLY BRACKETS
DOLLAR SIGN	RIGHT SLASH
LEFT PARENTHESIS	RIGHT PARENTHESIS
COLONS	SEMICOLONS
ASTERISKS	ACCENTS
AVERAGE WORD LENGTH	AVERAGE NO. OF SENTENCES

In addition to these features, counts of the following "stylistic features" as mentioned in [10] have been included in the dataset since the commonalities of these features are descriptive of the author behind the text.

3.2. TF-IDF

TF-IDF algorithm was applied on textual data to form a sparse matrix with large feature size (since every word accounts for an individual feature) and the frequency matrix of stylistic features was scaled down to 0-1. The aggregate of both these matrices was the dataset on which the model trained subsequently.

4. Training

4.1. Standard Models

In order to justify the inclusion of stylistic features, we initially trained the models without the same and evaluated accuracy scores. The scores significantly improved on inclusion of stylistic features. It was expected of SVM and Random Forest Classifiers to be able to generate clusters of different authors for all their written text on a generic dataset.

4.2. Deep Learning implementation

In this section, we list out the deep learning models we used in our project. We implemented sentence level LSTM, GRU and BiLSTM and article level LSTM, GRU and BiLSTM. For word representations we utilized GloVe word vectors of size 100 [7]. We utilized one of the most sophisticated models of glove.6B.100 where most of the words were represented.

Table 2. Results from Standard Models

HEIGHTMODEL	ACCURACY	PRECISION-RECALL
LINEAR SVC	84.75	0.88
RANDOM FOREST	92.2	0.95
MULTINOMIAL NB	87.4	0.98

5. Results

5.1. Standard Models

The results for training using some standard models viz. Linear SVC, Random Forests, and Multinomial Naive Bayes classifiers are tabulated in table 2.

5.2. Sentence Level GRU, LSTM and Bi-LSTM models

We utilized our C50 dataset and implemented deep learning at the sentence level. It was observed that the training f1-score always increases as you increase the number of epochs, but the testing score becomes constant after a certain number of iterations. An optimal number of epochs is thus chosen when the testing score stabilizes. The results are shown in Figures 1 and 2.

Figure 1. Training and Testing Accuracies vs Epoch

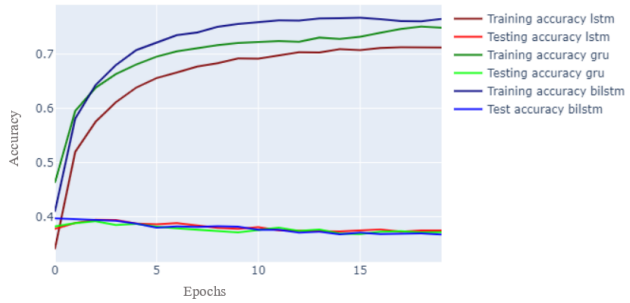
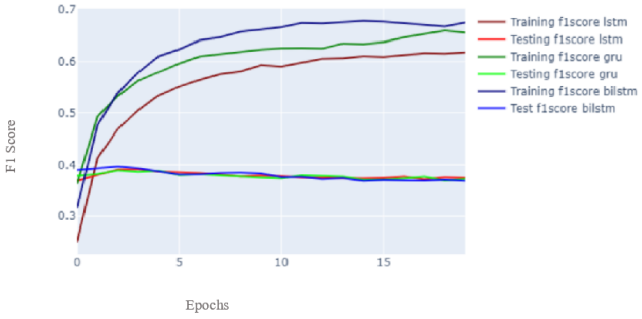


Figure 2. Training and Testing F1 scores vs Epoch



From Figure 1 it is seen that the test accuracy is too low compared to the training accuracy. It is also seen that GRU

training accuracy is higher than that of LSTM which is mainly because GRU is much more efficient than LSTM when the data set is small. This model of sentence level GRU, LSTM and Bi-LSTM has few models and many parameters and hence could lead to over-fitting therefore, we implemented article level methods.

5.3. Article Level GRU, LSTM and Bi-LSTM models

We utilized the same C50 data set to implement article level deep learning models. We plotted training accuracies, losses and F1 Score for the same and it as shown in Figure 3, 4 and 5.

Figure 3. Training and Testing Accuracies vs Epochs

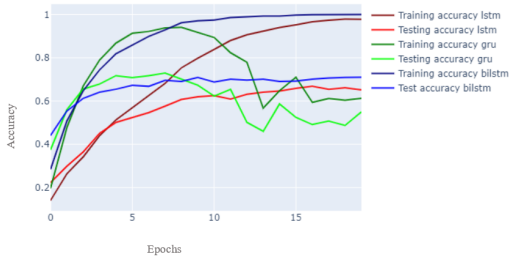


Figure 4. Training and Testing F1 Scores vs Epochs

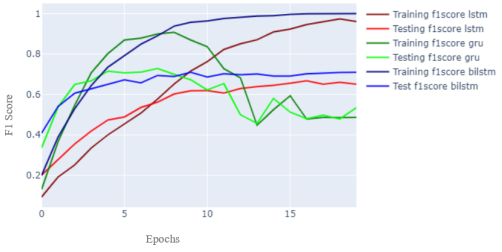
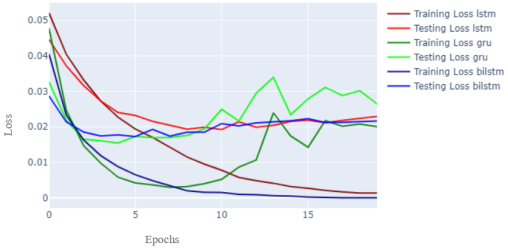


Figure 5. Training and Testing losses vs Epochs



For the above models we used a hidden size of 300 with a learning rate of 0.01. It is seen from the above figures that the testing accuracies for article level GRU, LSTM and Bi-LSTM models are much better than the sentence level implementations.

Table 3. Testing accuracies, loss and F1 score for all models for 20 epochs

MODEL	ACCURACY	F1 SCORE	LOSS
SENTENCE LEVEL GRU	37.14%	37.26%	0.066
SENTENCE LEVEL LSTM	37.42%	37.41%	0.0488
SENTENCE LEVEL BILSTM	36.7%	36.85%	0.058
ARTICLE LEVEL GRU	55%	53.45%	0.026
ARTICLE LEVEL LSTM	65.13%	65.09%	0.023
ARTICLE LEVEL BILSTM	71%	71%	0.021

From the above table, it is seen that the testing accuracies of all the three article level deep learning models are much better than the sentence level models. It can also be seen that the Bi-LSTM model is the best model with higher accuracy, F1 score and less loss. Refer to the Appendix for confusion matrices of these models.

6. Conclusion

In this project we implemented Linear SVM, Random Forest and Multinomial Naive Bayes models with Random Forest providing an accuracy of 92.2% but with lesser precision/recall score than Multinomial NB. Depending on the application, one metric can be weighed more than the other and the optimal model can be identified. It is also seen that, Article level deep learning models perform better than sentence level models. Article level Bi-LSTM models have the highest accuracy of around 71%. Better stylometric features could be implemented on our standard models and hence they have better accuracies.

7. Future Work

We would like to implement temporal relationship among words since our method ignores the temporal relationship and works only on averaging of word vectors. It is seen that GRU is not a good method compared to BiLSTM and LSTM models for this data set and hence we would like to implement various other deep learning methods like MvRNN (for example as implemented in [8]). We also want to implement metaphor detection methods utilizing similar models, this would help in creating a temporal relationship as mentioned above as implemented in [9]. This can also be used to detect plagiarism in various papers. We also believe that the implemented models can be utilized for various other applications like sentiment analysis and comparison of financial reports as mentioned in [1].

8. References

[1] Qian, Chen, Tianchang He, and Rao Zhang. "Deep learning based authorship identification." Department of Electrical Engineering, Stanford, CA (2017).

[2] Wang, L. Z. "News authorship identification with deep learning." (2017).

[3] A. M. Mohsen, N. M. El-Makky and N. Ghanem, "Author Identification Using Deep Learning," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, 2016, pp. 898-903, doi: 10.1109/ICMLA.2016.0161.

[4] M. Koppel and Y. Winter. Determining if Two Documents are by the Same Author. Journal of the American Society for Information Science and Technology, 65(1):178-187, 2014.

[5] Bagnall, Douglas. "Author identification using multi-headed recurrent neural networks." arXiv preprint arXiv:1506.04891 (2015).

[6] Lipton, Zachary C., et al. "Learning to diagnose with LSTM recurrent neural networks." arXiv preprint arXiv:1511.03677 (2015).

[7] J. Pennington, R. Socher, C. Manning, GloVe: Global Vectors for Word Representation, EMNLP, 2014.

[8] Socher, Richard, et al. "Semantic compositionality through recursive matrix-vector spaces." Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning. Association for Computational Linguistics, 2012.

[9] Sun, Shichao, and Zhipeng Xie. "Bilstm-based models for metaphor detection." National CCF Conference on Natural Language Processing and Chinese Computing. Springer, Cham, 2017.

[10] Calix, K., Connors, M., Levy, D., Manzar, H., MCabe, G., Westcott, S. (2008). Stylometry for e-mail author identification and authentication. Proceedings of CSIS research day, Pace University, 1048-1054.

9. Appendix

Figure 6. Confusion matrix for article level GRU

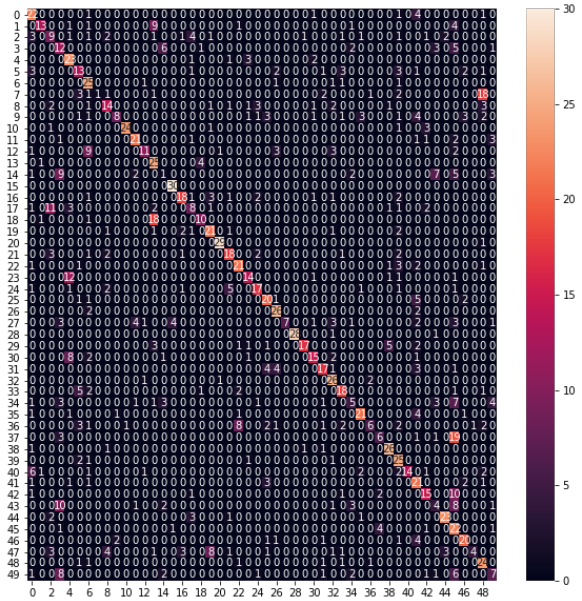


Figure 7. Confusion matrix for article level LSTM

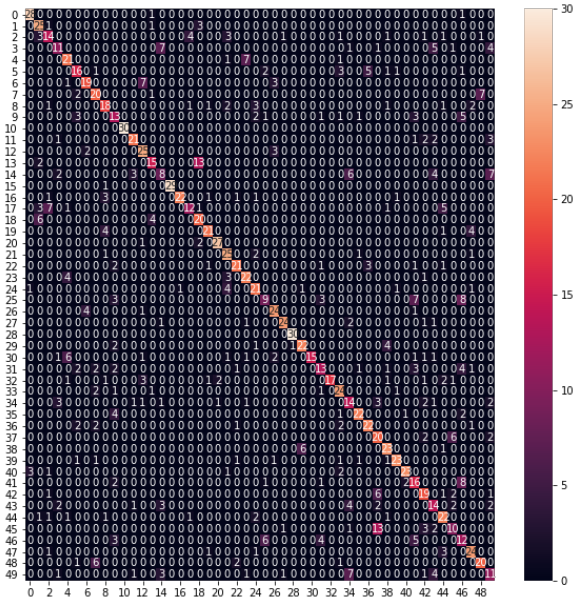
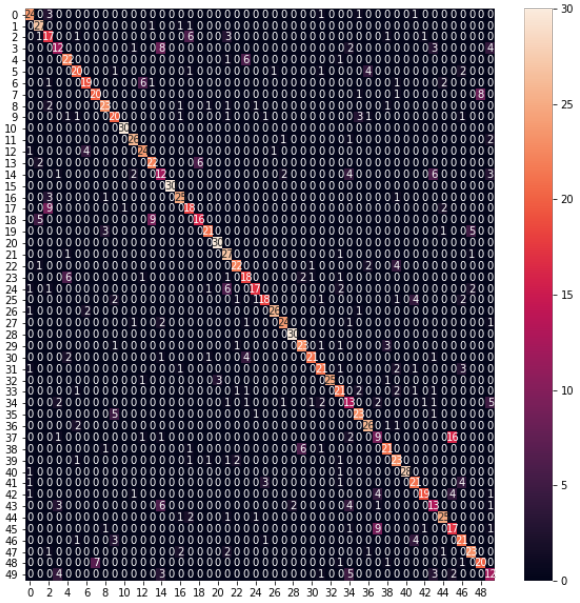


Figure 8. Confusion matrix for article level Bi-LSTM



Deep Learning based Authorship Identification

Presented by:

Arth Talati
Anirudh Sharma C A
Ranjani Narayanan

Video:
https://vimeo.com/415780627?ref=tw-share

Original Dataset
Reuters C_50_50

5000 Texts → 100/author

Analysis of dataset
Baseline approach

- **Low accuracy ~ 15 - 20%**

Data Pre-Processing

- **Removal of stop-words viz. 'a', 'is', 'are', etc.**
- **Retain words only present in less than 50% documents**
- **Retain words only if they are present in at least 5 documents**

TF-IDF implementation of word occurrences → sparse matrix

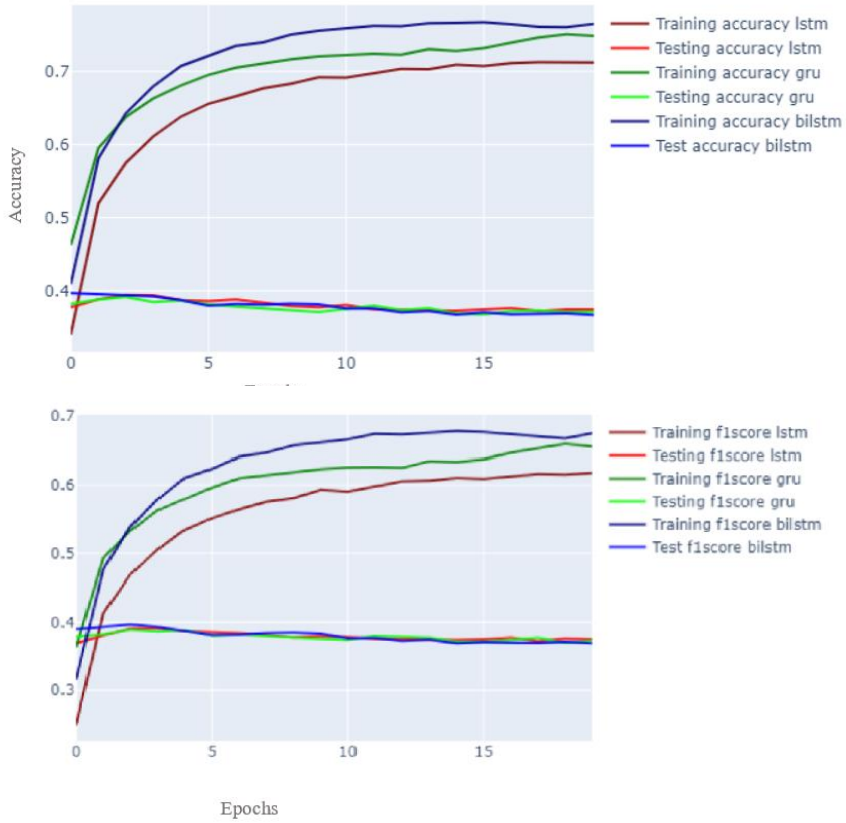
Stylometric Feature count

Full-stops	Apostrophes
Comma	Line Breaks
Question marks	White spaces
Exclamation marks	Ellipsis
Ampersands	Left curly brackets
Percentage sign	Right curly brackets
Dollar Sign	Right slash
Left Parenthesis	Right Parenthesis
Colons	Semicolons
Asterisks	Accents
Average word length	Average no. of sentences

Training and Testing datasets

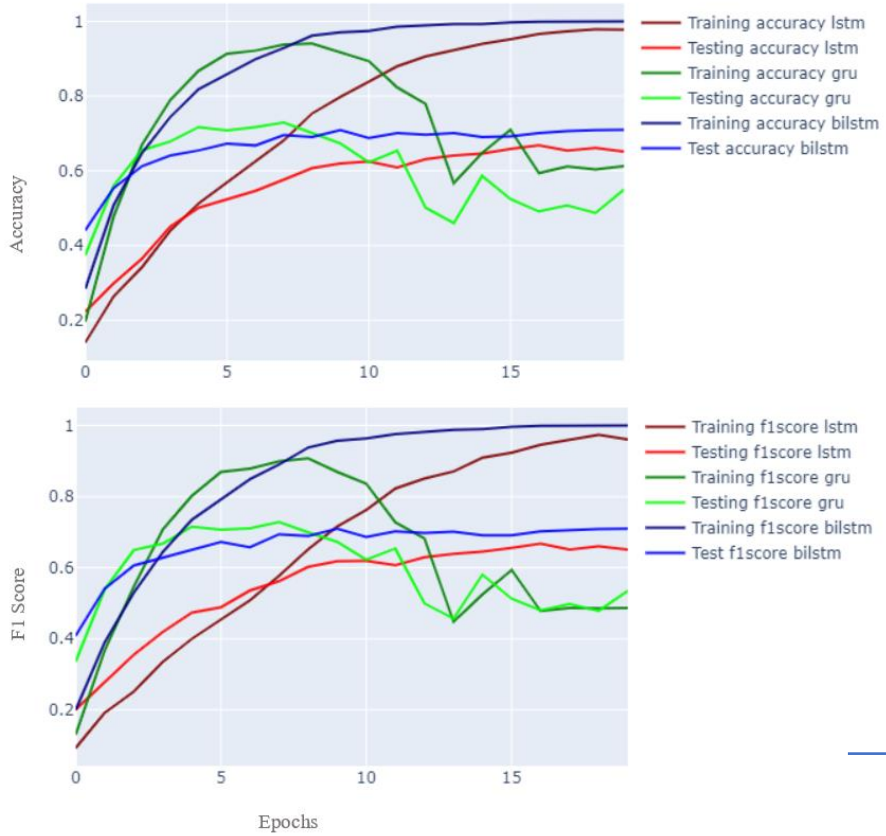
RESULTS

Sentence Level



Model	Test Accuracy	F1 Score
GRU	37.14%	37.26%
LSTM	37.42%	37.41%
Bi-LSTM	36.70%	36.85%

Article level



Model	Test Accuracy	F1 Score
GRU	55%	53.45%
LSTM	65.13%	65.09%
Bi-LSTM	71%	71%

Comparison with Standard Models

1. Standard models→ higher accuracies and P/R scores

2. Article level deep learning models perform better than sentence level models

3. Article level Bi-LSTM models have highest accuracy

4. Stylometric data Standard Values→ yield better accuracies

Model	Test Accuracy	Precision/recall
Linear SVC	84.75%	0.88
Random Forest	92.20%	0.95
Multinomial NB	87.40%	0.98