Authorship Attribution with Document Encodings and Neural Networks

1st Miles Baer
Dept. of Computer Science
Middle Tennessee State University
Murfreesboro, United States
mtb3x@mtmail.mtsu.edu

2nd Robert Smith
Dep. of Computer Science
Middle Tennessee State University
Murfreesboro, United States
rws2p@mtmail.mtsu.edu

3rd William Cope
Dept. of Computer Science
Middle Tennessee State University
Murfreesboro, United States
wrc2t@mtmail.mtsu.edu

4th Charles Johnson
Dep. of Computer Science
Middle Tennessee State University
Murfreesboro, United States
cwj2z@mtmail.mtsu.edu

5th Nathaniel Boyer
Dept. of Computer Science
Middle Tennessee State University
Murfreesboro, United States
njb3n@mtmail.mtsu.edu

***Abstract*—Our goal is to determine authorship of short texts using neural networks and document level encodings. Neural networks (NNs) are capable of finding correlations between inputs and their expected outputs. This allows them to perform quite well at classification tasks such as authorship attribution where the inputs would be documents and the classifications their respective authors. Currently, Word2Vec and Doc2Vec are popular methods used to build vocablary models. The models can then be used to encode words and documents respectively into fixed sized vectors containing floating point values. We propose various methods that utilize Word2Vec and Doc2Vec as a means of building document level encodings that retain the stylistic elements of the author. Primarily, we used three different NN architectures inspired by previous work in authorship attribution and sentiment analysis research. With each network we tested multiple methods of encoding amazon reviews, treating each review as a document whose classification was the author. Testing shows that none of the attempted document embeddings for short texts are able to outperform previous methods.**

***Index Terms*—neural networks, convolution, Word2Vec, Doc2Vec, document encodings, authorship attribution**

I. Introduction

Text-based classification has been used for many different things. A lot of it has to do with semantics checking and the like. We are trying to find unique stamps for an individual author based on they way that they write stylisticly. We originally surmised that Word2Vec along with Doc2Vec could find these similarities and build a unique profile for an author on a per document basis. The nueral networks would try to use this unique vector data to build a profile for an individual author. We have referenced a paper [reference here] using tweets from twitter as input. The paper is using n-grams for single words, making the window or filter surrounding characters instead of groups of words. Our approach will be to use word groups instead of looking at individual characters in a single word or groups of words. Our custom model uses Word2Vec on individal words in a document and sums the corrosponding word vectors to fake a Doc2Vec model. Word2Vec uses (CBOW) while Doc2Vec uses (DBOW), so there is a major distinction here. We want to know if these distinctions will matter when being trained in various neural networks. Sampling also could have different effects. In our models, some runs have normal training data, while the others has been oversampled to try and improve training and testing accuracy. In terms of nueral networks, massive nets proved to be very slow at training and unsuccessful in giving good similarity predictions, [1].

II. Background

Our project is based loosely off a previous paper doing research using twitter. Small texts is the main similarity between the two in addition to using CNN's. They took reviewers with a minimum of 1000 tweets and began to classify them. [2].

III. Methods

We used a combination of gensim and spacy to get our encodings for Word2Vec and Doc2Vec. We have used Google's pretrained model and even trained our own custom models to produce the word vectors. Our custom model was much smaller than Googles. We first organize the documents (representing a user and their review) and place the auther and the review seperated by a tab on a single line. This is done for every review to make parsing easier. Sorting is done by author to make splitting the data into training and test easier. We then load up the two models and run our data through them. The output is the author and their encoded vector of words for that current document seperated by a tab on a single line for all documents. Now the data is prepped and ready to strung through the 3 nueral networks we have made. Once loaded they are split into appropriate training and test sets and converted into numpy arrays for use in the net.e [3].

IV. Results

The results come from using three basic nueral networks. Each was tested with 3 authors and 35 authors. And on each

of these tests we used normal sampling and oversampling to try and see how that affects training on these similarities of authors. We also tested it on 500 and 5000 authors, but there was no improvement. It actually negatively impacted the results. [4].

V. Discussion

In general, our tests along with our neural networks, proved to be mostly unsuccessful at building a unique feature-set for classifying indivudual authors. The best result that was acheived (referenced above) was only acheivable using 3 authors. As the number of authors increased, the accuracy decreased sharply. Overfitting was a huge problem proving to be our biggest downfall. After around 5-10 epochs, the loss goes down to zero (or very close to) and the accuracy shoots up to 1. Along with that, Word2Vec and Doc2Vec are not the best at picking up stylist differences in authorship. They adept at finding semantic similarites though. Maybe if we had increased the width of these word vectors, they could've been more unique. Removing all alpha-numeric characters may have affected it as well. Maybe knowing the context via punctuation would've proven useful to training the model in Word2Vec and Doc2Vec. [5].

References

[1] L. D. Trang and Z. Mebkhout, "Variétés caractéristiques et variétés polaires," *C. R. Acad. Sc. Paris*, vol. 296, pp. 129–132, 1983.

[2] J. Birançon, P. Maisonobe, and M. Merle, "Localisation de systèmes différentiels, stratifications de Whitney et condition de Thom," *Invent. Math.*, vol. 117, pp. 531–550, 1994.

[3] A. Parusiński and P. Pragacz, "A formula for the Euler characteristic of singular hypersurfaces," *J. Alg. Geom.*, vol. 4, pp. 337–351, 1995.

[4] K. Elissa, "Title of paper if known."

[5] R. Nicole, "Title of paper with only first word capitalized," *J. Name Stand. Abbrev.*