# Author Attribution with CNN's

**Dylan Rhodes**
Department of Computer Science
Stanford University
dylanr@stanford.edu

## Abstract

In this report, the results from my CS224D final project are given and explained. The project was based on the application of relatively new neural network architectures, namely convolutional neural networks over word embeddings, to the task of authorship identification. The problem was posed as a classification task and models were evaluated over two datasets, a baseline of my own collection and a competition dataset with pre-existing, published results. Despite robust results on the baseline dataset, the CNN architecture failed to outmatch the best competition submission, although it did outscore most of the competitors' submissions.

## 1 Introduction

Authorship attribution is a well-studied problem among NLP researchers which dates back to the earliest attempts at quantitative analysis of text documents. The goal is to match anonymous text with its author via some similarity measurement learned from labeled text written by the same person. There are several common real world applications including plagiarism detection, identification of authors of threats, verification of suicide notes, computer forensics, and intelligence, where it is widely used and studied. In this paper, the closed-group formulation of the attribution problem is studied, which asserts that all authors of documents in the test data appear in the training corpus as well. This assertion allows one to pose the problem as a classification task over natural text without concern for modeling the case of an unseen author.

## 2 Related Work

### 2.1 Author Identification

The field of authorship attribution began sometime in the late 19th century with the first attempts to quantify writing style. Interest in the field was primarily generated by the desire for more objective justifications for the authorship of disputed works by authors such as Shakespeare and Bacon. Statistical methods were dominant in the field by the time of the early studies of Zipf [13] and Yule [12] of relative word frequencies, which discovered that the frequency with which a word appeared in a text could be approximated with a Poisson distribution. A classic analysis on the authorship of the Federalist Papers by Mosteller and Wallace in 1964 via Bayesian analysis of frequency counts of the most commonly used words proved heavily influential on the field and lead to decades of the construction of 'stylometry,' essentially detailed, manual feature engineering[10]. Through the late 1990's, most research on authorship attribution focused on new feature proposals, most of which have been since disregarded. Research in this time period suffered from a lack of open benchmark datasets, uncertain ground truth labels of studied texts (generally mis-or-unattributed works), failure to control results for topic, and small datasets and sets of candidate authors. These fundamental issues culminated in a general lack of progress in the field besides very specific cases like the Federalist Papers.

Since then, the field of authorship attribution has benefited from the rise of large-scale web-derived datasets with objective evaluation metrics as well as a new set of statistical models which can more effectively handle large and sparse corpora. In addition, the focus of the field has shifted from strictly literary analysis to some of the more immediately relevant ends listed above [1]. Several well-known competitions over topics like plagiarism detection, author identification, and author profiling have arisen with well-curated datasets, large numbers of participants, and better comparison of successful and unsuccessful methods. These trends, as well as increased support by institutions including law enforcement and government, have led to a resurgence of the subfield and greatly improved results on open benchmarks. The two most commonly cited authorship competitions are the Ad-hoc Authorship Attribution Competition and the PAN (International Workshop on Plagiarism Detection, Author Identification, and Near-Duplicate Detection), which have been run in 2004 and annually from 2007 respectively [4] [11]. Statistical methods including Naive Bayes, compression models, and various distribution similarities have been used to analyze n-gram counts for authorship attribution. Most recently, a plethora of models more familiar to machine learning practitioners than linguists such as support vector machines, latent Dirichlet allocation, decision trees, and neural networks have been applied to different types of word embeddings with success [4] [6] [11]. To my best knowledge, researchers have yet to formally evaluate the capabilities of convolutional network models to the problem, but with the PAN 2015 conference due to commence in early September, it is only a matter of time.

### 2.2 CNN's for NLP

Although convolutional neural networks have been tested and applied to image classification and detection problems since the 1990's, they were generally disregarded by most researchers until Alex Krizhevsky and other members of Geoffrey Hinton's University of Toronto vision lab achieved a significant breakthrough in classification scores with their submission to the 2012 ILSVRC challenge. AlexNet, a deep (at the time) convolutional neural network with innovations such as dropout regularization and the rectified linear unit nonlinearity, rekindled interest in deep learning within the field and ushered in a whole class of research based on the use of CNN's as feature transforms for image regions[7].

The cross-application of this new interest to natural language datasets took some time. The main obstacles to the direct application of CNN's to language were the variable length nature of text documents and the relative difficulty of resizing them as compared to images. It was also generally unclear if these models could perform as well as already existing network architectures such as recurrent and recursive networks. Towards a solution to the variable-length problem, there have been several proposals including max-over-time pooling, k-max pooling, and dynamic k-max pooling [2] [5]. These operations ensure a constant-sized output for convolutional layers at a given depth in the network, allowing the inclusion of more traditional types of layers and clear means of applying back-propagation. The max-over-time pooling operation, which is employed in this project, is described in detail in the next section. With these tools, researchers found that these types of models could be trained to produce state-of-the-art results on a variety of tasks including sentiment analysis, named entity recognition, and question classification among others over existing sets of word embeddings [6] [5] [2].

## 3 Approach

The goal of my project was to examine the application of these relatively new types of sentence classification models to the task of authorship attribution. In this section, I describe in detail the architecture of the models ultimately trained for the task as well as the datasets over which they were evaluated. The approach was heavily influenced by that of Kim 2014 and Razavian et al. 2014, which illustrated that pre-trained feature extractors could be effectively cross-applied to different tasks in vision and language processing [6] [9].

### 3.1 Quantifying Words and Sequences

The initial problem of word vectorization has been studied from many angles over the last decade. A plethora of unsupervised models have been proposed, most of which rely on co-occurrence frequencies and the application of an auto-encoder. It would not be worthwhile to describe this body of research in its entirety here, but briefly, some well-known and currently used word vectorization formulations include the skip-gram model, continuous bag of words model, and GloVe vector model. For the purposes of this project, a set of word vectors trained via the skip-gram method and negative sampling were employed for vectorization. The word vectors selected had dimensionality of three hundred and had been derived from co-occurrence in a dataset of articles from Google News. They are available for public use online [8]. Words which were not included in this set were initialized randomly with a small Gaussian.

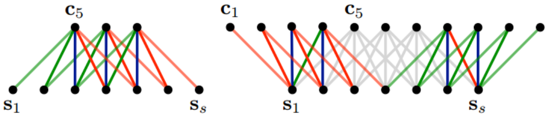

Figure 1: Illustration of narrow (left) and wide (right) convolution for a 5-gram filter over a sequence of eight words.[5]

In order to encode sequences, rather than words, a simple concatenation operator was applied. This is the standard approach for convolutional models, although other types of models have employed summation or mean operators to produce sequence representations. The concatenated word vectors were padded with zeroes on either side so that the convolutional operation would be wide rather than narrow, which prevents the network from focusing on center words. A graphical comparison of wide and narrow convolutions over word vectors is illustrated in Figure 1.

### 3.2 Max-over-time Pooling

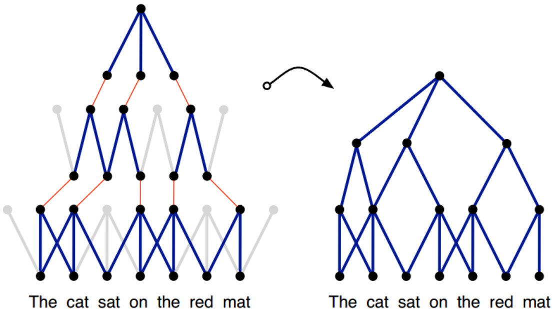

Figure 2: Two layer CNN with three 3-gram filters and two 2-gram filters on the first layer. The left diagram shows max-over-time pooling selections in red and unselected filter output nodes in gray. The right diagram depicts the resulting network graph.[5]

The max-over-time pooling function is a key component of the convolutional networks examined in this paper. It reduces the outputs of a filter bank applied to text segments of varying length to a constant-sized vector representation. First introduced by Collobert et al in 2011 as a means of constructing their general convolutional network for language analysis, it simply takes the maximal output of each filter as applied to all n-grams in the sequence under consideration as that filter's output for the sequence. This may seem like an overly destructive operation as it foregoes explicitly encoded knowledge about n-gram order and duplication, but it actually produces a robust descriptor for the sequence, as illustrated by my experimental results. Recent work by Kalchbrenner et al. like the dynamic max-k pooling operator offers a means of retaining this information in an explicit form, which may be a fruitful avenue for future analysis.

### 3.3 Architecture

The actual architecture of the convolutional networks applied here is closely derived from the work of Collobert et al 2011 with some small modifications including dropout and the use of rectified linear units instead of the hyperbolic tangent function [2]. The input layer is a concatenated sequence of word vectors padded with zeroes as a start and stop token, as previously described. The next layer is convolutional and includes three filter banks of one hundred filters each over 3-grams, 4-grams, and 5-grams. The output of this convolutional layer is normalized to a constant size via the max-over-time pooling operation and passed through a rectified linear unit non-linearity function. As a regularization step, dropout is applied to this non-linear layer, and finally, classification is performed via logistic regression.

The network is trained via backpropagation and the AdaGrad update rule, which has been shown to decrease convergence time over vanilla gradient descent. Backpropagation is well defined for all of the operations previously described, so this step is fairly straightforward. Ultimately, for the competition dataset backpropagation was performed over all layers of the network including the input layer, a decision which was motivated by some experimental results presented below.

### 3.4 Datasets

Over the course of this project, I performed experiments on two datasets. The first was of my own design and primarily intended for validation of the model and hyperparameters in a general sense. The second was taken from the PAN 2012 author identification challenge.

#### 3.4.1 Canada Dataset

| **Title** | **Author** |
|---|---|
| Canada and the Canadians: Vol. 1 | Sir Richard Henry Bonnycastle |
| Canada and the Canadians: Vol. 2 | Sir Richard Henry Bonnycastle |
| Canada | John George Bourinot |
| Canada Under Britain | John George Bourinot |
| Country Life in Canada in the Last Fifty Years | Canniff Haight |
| Pioneers in Canada | Sir Harry Hamilton Johnson |
| Makers of Canada: Champlain | E.L. Dionne |
| Makers of Canada: Laval | Adrien LeBlonde |

Table 1: Simple literary dataset sources

The self-collected dataset was a collection of public domain books sourced from Project Gutenberg [3]. They were selected for homogeneity of topic and time of publication so that the network would be forced to primarily rely upon writing style disambiguation rather than more shallow features. The full list of titles and their authors is presented in Table 1. As you can see, all of the titles concern life in Canada during the 19th century and were all published at roughly the same time. The dataset comprises eight books written by six separate authors and is slightly unbalanced in favor of the authors with more than one included work, although not extremely so.

#### 3.4.2 PAN 2012 Contest

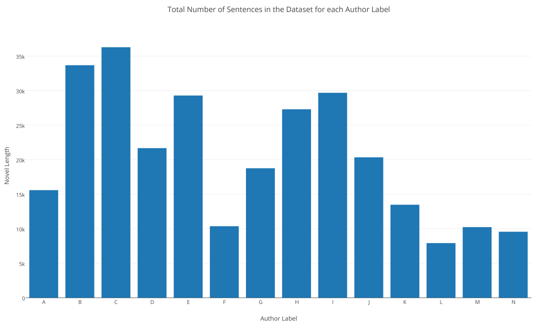

Figure 3: Distribution of sentence counts within the train and test dataset for all fourteen authors.

The second dataset over which I conducted authorship attribution experiments was derived from the 2012 PAN author identification challenge. The dataset consists of twenty-eight relatively unknown works of science fiction in English written by fourteen separate authors. These twenty-eight works are evenly distributed into a training set and test set which each include a single work from each author respectively. The training and test splits from the original contest were preserved for my experiments so that the results could be directly compared. The contest dataset shares many helpful characteristics with the Canada dataset. Most importantly, the documents concern a single topic (or genre at the very least), so they are not trivially separable based on the inclusion of very specific words, a problem which plagued early authorship identification analyses. Like the Canada dataset, it is somewhat unbalanced in terms of sequences attributed to each author, although not overly so. A graphical comparison of the prevalence of each author's work in the training data is given in Figure 3.

# 4 Experiments

## 4.1 Canada Dataset

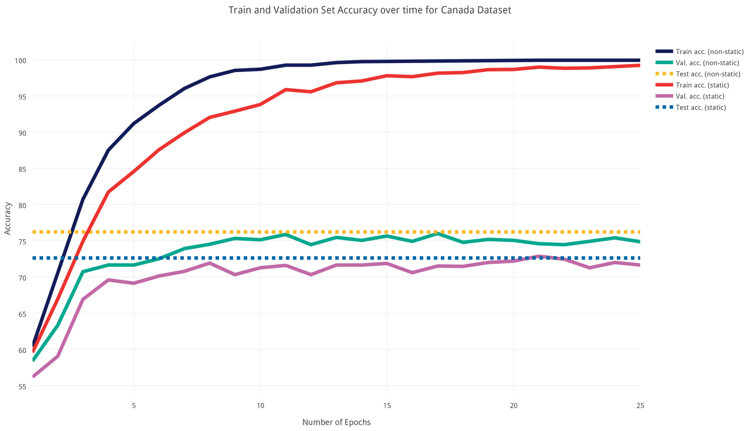

Figure 4: Accuracy curves for training and validation splits as well as final test accuracy for both a model with fine tuning and without. The document may be zoomed to enlarge the labels.

The purpose of including the Canada dataset in my project was mainly as a check that my model had been correctly implemented and that the hyperparameter settings fell within a reasonable range of values. On this front, the performance of the model on the dataset was reassuring in that it easily surpassed the level of a random and even a fairly robust classifier. With this dataset, I sought to validate the choices made for the subsequent, more important experiments on the PAN data. One decision I was eager to validate was whether to fine tune the word vectors themselves or just to learn the network on top of them. Towards this end, I trained two networks over the dataset, one with backpropagation enabled only up to the convolutional layer and another through the input layer as well.

To evaluate the performance of the networks, I split the sentences in the dataset into three groups. One tenth was randomly assigned to a test set and another tenth was assigned to a validation set. The remaining sentences were used to train the network over a period of twenty-five epochs. The network architecture was identical to that outlined in the architecture section above, including three banks of filters of one hundred each over 3-grams, 4-grams, and 5-grams. The quantitative results of this experiment are illustrated above in Figure 4. This chart gives an accuracy curve for the training and validation sets as well as a dotted line for the final test accuracy. As you can see, the model with full backpropagation enabled (denoted non-static in the chart) outperformed the model without fine tuning by roughly four percent. The final test accuracies for the static and non-static models were 72.67% and 76.38% respectively. Based on this result, I decided to train my final PAN 12 model with full backpropagation and fine tuning of the word vectors.

### 4.2 PAN 2012 Contest

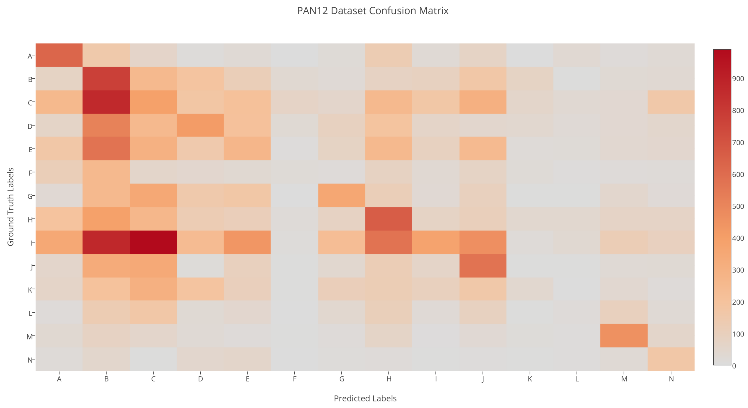

Figure 5: Confusion matrix for the final model over the PAN12 test dataset. The document may be zoomed to enlarge the labels.

Results on the PAN 2012 dataset were also promising. Initially, a network with identical hyperparameter settings to that described above was trained and evaluated on the dataset. However, I noticed that it suffered from extreme overfitting from very early on and suspected that it could be improved with greater regularization. Ultimately, the dropout probability was increased from $p = 0.5$ to $p = 0.75$ in my final model, which decreased overfitting long enough to improve sentence classification accuracy by about three percent. The architecture and filter bank sizes were equivalent to those of the originally described model. A heatmap of the confusion matrix generated by the final network over the test set is given in Figure 5.

As you can see above, the network is already a robust classifier for PAN12 data. However, there is a tendency to over predict the classes which have the largest representation in the dataset. This implies that the network learns a bias on its own, which is somewhat interesting in and of itself. Nonetheless, this tendency negatively impacts my results as compared to the PAN 12 contest, since it is evaluated on the basis of document author prediction accuracy, not over sentence fragments. To use these classification results directly as a bagged predictor for the author of each document would be suboptimal, since many of the documents would end up simply classified as written by one of the authors with highest representation in the dataset.

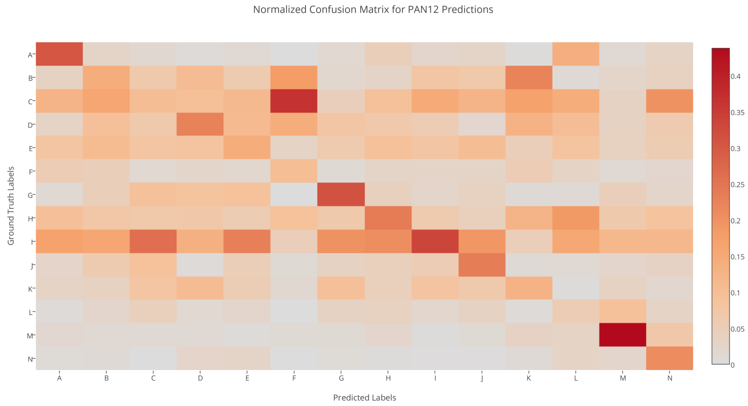

Figure 6: Class normalized confusion matrix for the final model over the PAN12 test dataset. The document may be zoomed to enlarge the labels.

This problem can be resolved by the application of Bayes' rule. If one takes the prior to be the proportional representation of each author's work in the predictions and the normalizing factor to be constant, since each author has an equivalent chance of writing a given document, one can normalize the votes of the network to alleviate the class imbalance problem. A post-normalization heatmap is illustrated in Figure 6. Taking a maximum over these normalized prediction votes rather than the pre-normalized counts correctly classifies twelve of the fourteen documents as opposed to the nine originally correctly classified and assumes no additional knowledge of the test labels.

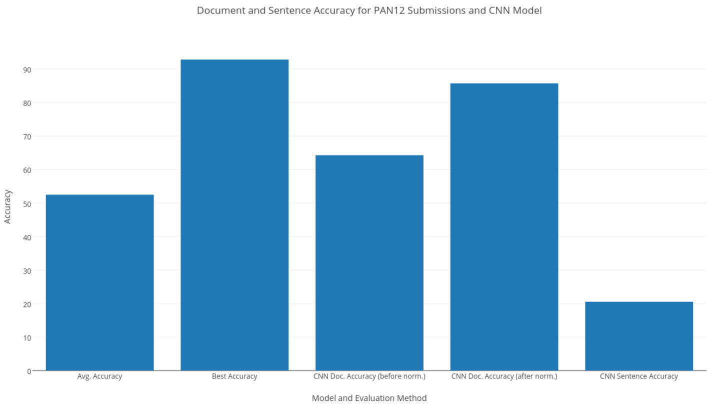

Figure 7: Comparison of the accuracy achieved on the PAN12 author identification dataset by various groups and models.

Twelve out of fourteen documents is a robust result for the PAN12 author identification dataset. In fact, although the top submission correctly attributed thirteen of the fourteen test documents, the average accuracy of submissions for the contest was only 7.35 out of fourteen. The final accuracy rate of sentence fragments contained in the test set was $20.52\%$, about three times the accuracy of a random classifier. I consider this to be a promising result, since authorship attribution at the single sentence level is an extremely difficult task for even humans. An illustration of competitors' average results on the competition, the best result, my own before and after normalization, and the sentence fragment accuracy is given in Figure 7.

## 5 Conclusion

Overall, I was satisfied with the performance of the CNN models on the authorship attribution task. They posted a robust result which easily surpassed that of most of the competitors in the original contest, even before correcting for class imbalance in the bagged prediction. The CNN models presented here employ a relatively simple architecture; implementing a DCNN in the spirit of Kalchbrenner et al, which can encode long-range dependencies as well as spatial and duplication information about n-grams found within the word sequences would likely produce even higher accuracy scores.

There are also several avenues of exploration which I did not have time to examine. The initial catalyst for choosing my project topic was to evaluate a model for source code attribution. There is some preexisting work on this subject, but not for the models examined here [1]. Ultimately, I ran out of time to test the efficacy of these models on that task, mainly because of the additional burden of training word embeddings for source fragments. Another useful extension would be a robust means of modeling out-of-set authors, which would allow one to apply these types of models to the open-group author identification task rather than the more restrictive closed-group task presented here.

Regardless of these limitations and areas of further work, I enjoyed the implementation and evaluation of this project and look forward to reading about the results from PAN 2015.

## References

[1] Upul Bandara and Gamini Wijayarathna. Source code author identification with unsupervised feature learning. *Pattern Recognition Letters*, 34(3):330–334, 2013.

[2] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537, 2011.

[3] Michael Hart. *Project Gutenberg*. Project Gutenberg, 1971.

[4] Patrick Juola. Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3):233–334, 2006.

[5] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

[6] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[8] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[9] Ali S Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*, pages 512–519. IEEE, 2014.

[10] Efstathios Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556, 2009.

[11] Efstathios Stamatatos, Walter Daelemans, Ben Verhoeven, Martin Potthast, Benno Stein, Patrick Juola, Miguel A Sanchez-Perez, and Alberto Barrón-Cedeño. Overview of the author identification task at PAN 2014. *analysis*, 13:31, 2014.

[12] G Udny Yule. On sentence-length as a statistical characteristic of style in prose: With application to two cases of disputed authorship. *Biometrika*, 30(3-4):363–390, 1939.

[13] George Kingsley Zipf. Selected studies of the principle of relative frequency in language. 1932.