Duquesne University
Duquesne Scholarship Collection

Electronic Theses and Dissertations

Spring 2013

Authorship Attribution Through Words Surrounding Named Entities

Julia Maureen Jacovino

Recommended Citation

Jacovino, J. (2013). Authorship Attribution Through Words Surrounding Named Entities (Master's thesis, Duquesne University). Retrieved from https://dsc.duq.edu/etd/694

AUTHORSHIP ATTRIBUTION THROUGH WORDS SURROUNDING NAMED ENTITIES

A Thesis

Submitted to the McAnulty College and Graduate School of Liberal Arts

Duquesne University

In partial fulfillment of the requirements for

the degree of Masters of Science in Computational Mathematics

By

Julia Jacovino

May 2013

AUTHORSHIP ATTRIBUTION THROUGH WORDS

SURROUNDING NAMED ENTITIES

By

Julia Jacovino

Approved April 2, 2013

Patrick Juola Ph.D.
Associate Professor
Department of Mathematics & Computer Science

Donald Simon Ph.D.
Director of Graduate Studies
Department of Mathematics & Computer Science

James Swindal, Ph.D.
Dean
McAnulty College and Graduate School of Liberal Arts

Eric Ruggieri Ph.D.
Assistant Professor
Department of Mathematics & Computer Science

ABSTRACT

AUTHORSHIP ATTRIBUTION THROUGH WORDS SURROUNDING NAMED ENTITIES

By

Julia Jacovino

May 2013

Dissertation supervised by Patrick Juola, Ph.D., Associate Professor

In text analysis, authorship attribution occurs in a variety of ways. The field of computational linguistics becomes more important as the need of authorship attribution and text analysis becomes more widespread. For this research, pre-existing authorship attribution software, Java Graphical Authorship Attribution Program (JGAAP), implements a named entity recognizer, specifically the Stanford Named Entity Recognizer, to probe into similar genre text and to aid in extricating the correct author. This research specifically examines the words authors use around named entities in order to test the ability of these words at attributing authorship.

DEDICATION

This thesis is dedicated to my parents, Gary and Maureen Jacovino, who have supported me throughout this research and my continuing education.

It is also dedicated to Dr. Patrick Juola who inspired my interest in the research of authorship attribution.

And to Mary Kay Wisniewski, for her guidance and love of education.

TABLE OF CONTENTS

LIST OF TABLES

Page

Chapter 1: Introduction

Through stylometry, the study of linguistic style and its applications, one can attribute authorship to anonymous or disputed texts. There are academic, legal and literary applications of authorship attribution, varying from the question of the authorship of Abraham Lincoln's works to forensic linguistics.

People use language in a multitude of ways. Writing styles differ depending on whoever composes a text. Authors express the same statement using different words that produce similar meanings. It is possible that an author of a text uses certain words more frequently in their writing than other authors.

This research was designed to take one aspect of possible authorship attribution – the frequencies of words used before and after named entities/proper nouns – and test its ability to identify the correct author. This addressed two questions: 1) Is it possible the words surrounding proper nouns can be written in different styles depending on an author? and if so 2) Can we perform statistical analyses on the frequency to attribute authorship? When implementing the Stanford Named Entity Recognizer with the Java Graphical Authorship Attribution Program, we can determine the amount of usage a word receives from an author to verify if another text receives similar usage amount by applying statistical analyses to both texts.

Chapter 2: Background

2.1 Authorship Attribution

Authorship attribution is a process seeking to identify correct authorship of a document based on an author's stylometry. Specifically looking at the words an author chooses to use in their writing can show characteristics of that author and their other documents. Scholars have utilized this process to determine unknown, disputed, and forged texts by quantitatively measuring an author's style. Modern statistics allow innovative approaches to determine correct authorship.

Authorship attribution has been ongoing for many years. One early example of authorship attribution stems from *The Federalist Papers*. *The Federalist Papers* are a collection of 77 political essays written between 1787 - 1788 and published in various newspapers under the pseudonym 'Publius.' The true authors of these works were Alexander Hamilton, James Madison and John Jay. Later in life the three authors disclosed which of the articles they wrote but their accounts differed. Five of the essays were attributed to John Jay, 43 to Alexander Hamilton and 14 to James Madison. Three of these essays were jointly written. Twelve of the essays were disputed between the authors.

Frederick Mosteller and David Wallace studied the essays in the early 1960s. They hand-picked 30 function words to analyze statistically. Function words are words whose purpose is to indicate grammatical relationship in a sentence rather than convey lexical meaning. Examples of function words are conjunctions, prepositions and grammatical articles. Looking specifically at the use of an author's function words in documents, they can quantitatively measure the frequencies of these words per author and compare these frequencies to the disputed texts. For example, Mosteller and Wallace noticed Madison used the word "*by*" between 11 – 13 times per 1,000 words and never used "*by*" less than 5 times. On the other hand, Hamilton used the word "*by*" between 7 – 9 times per 1,000 and never more than 13. With this and other information as well as statistical analyses, Mosteller and Wallace were able to attribute probable authorship to all of the disputed *Federalist Papers* (Mosteller & Wallace, 1963) .

Since Mosteller and Wallace's study an increasing awareness of authorship attribution has occurred. With the development of computers and modern statistics, scholars have developed computer-based authorship attribution programs. Computer-based authorship attribution allow users to use innovative techniques to analyze their authorship attribution problems.

A great deal of research still exists in authorship attribution. This process can be perfected as new ideas arise. As previously stated, this research investigates the ability to examine the frequencies of the words used before and after named entities. When looking into sentence structure, one notices the frequencies of function words that surround named entities. This research now embarks on the process of extracting and evaluating the words surrounding named entities.

2.2 JGAAP

JGAAP, which stands for Java Graphical Authorship Attribution Program, is a Java-based, modular, program for textual analysis, text categorization, and authorship attribution i.e. stylometry / textometry. JGAAP is intended to be used to tackle two different problems, "firstly to allow people unfamiliar with machine learning and quantitative analysis the ability to use cutting edge techniques on their text-based stylometry / textometry problems, and secondly to act as a framework for testing and comparing the effectiveness of different analytic techniques' performance on text analysis quickly and easily" (EVL Labs, 2010). This software program allows for research development and implementation of a named entity recognizer with an easy-to-use interface for the user.

JGAAP has several analysis methods built in. There are 18 different types of methods and 3 of those methods use distance functions. If choosing an analysis method that uses the distance functions, there are 25 different distance functions. This makes over 90 different analysis methods built in.

For this research, 3 different analysis methods were chosen to find the given event feature and make comparisons. They were WEKA Sequential Minimal Optimization (SMO), Centroid Driver: Alternate Intersection Distance and Centroid Driver: Cosine Distance. The Centroid Driver analysis method uses two different distance functions.

WEKA SMO was created by WEKA and implements John C. Platt's sequential minimal optimization algorithm developed in 1998 for Microsoft. This algorithm is used for training a support vector classifier using polynomial or RBF kernels (Frank, Legg, & Inglis). The SVM (Support Vector Machine) algorithm is summarized by the Evaluating Variations in Language Lab (EVL Lab) as follows:

> A statistical analysis technique which generates a separator to divide the document space into several regions, each corresponding to a specific author. That is, the set of documents is embedded in a high-dimensional space based on the features extracted in the Event Set. SVM is then used to generate a

separating hyper-plane in this space, or some higher-dimensional space in which the data may be linearly separable, based on the training data (the set of documents with known authors). Unknown documents are then embedded into the same space, and an authorship label is assigned based on which side of the hyper-plane the unknown document is placed. The transformation from the document space to a higher-dimensional linearly separable space is defined implicitly within the kernel function. A kernel function is essentially a distance metric in some high-dimensional space. It takes inputs in a low dimensional space and calculates their distance within the higher-dimensional space without actually performing the projection to this higher-dimensional space (EVL Labs, 2010).

The Centroid Driver computes one centroid per author. This is the opposite of the nearest neighbor approach that takes the closest matching document and assumes the author of the matching document to be the author of the unknown. The Centroid Driver instead finds the average relative frequency of events (features) over all documents provided by known authors. This produces several centroids; one for each author. The unknown/disputed document is assigned the same author who has similar frequencies of an event based on the centroid of an author, not individual works. The Centroid Driver uses both alternate intersection and cosine distances when finding the centroid for this research.

These methods of analysis were chosen to be used in JGAAP since they have previously proven to be the best current methods to attribute authorship.

Multiple event drivers are also built-in to the JGAAP framework. Three were chosen to be utilized in this research to compare to the WordsBeforeAfterNamedEntities event driver. WordsBeforeAfterNamedEntities event driver was specifically developed for this research. The three chosen comparison event drivers are Words, Char4Grams, and Char8Grams.

Looking more specifically at the aforementioned comparison event drivers we see exactly what they return when running in the JGAAP framework. The Words event driver extracts single words from a text as features. A word here is defined as a “maximal sequence of whitespace-delineated characters. That is, any string of characters without whitespace between them will be considered a word. Hence, words can contain punctuation, numerals, etc. Note that the whitespace characters themselves are not considered words or parts of words” (EVL Labs, 2010). The character event driver extracts individual characters (letters, numbers, punctuation, white space, symbols etc.). In this research, the character event driver works in conjunction with the n-grams event driver. Together, we get the Char4Grams and Char8Grams event drivers. For these two specific CharNGrams, JGAAP first extracts the single character and adds it to an event set. From this event set, the n-gram event driver runs through the event set and separates the characters *n* at a time. For example, consider the sentence “Mike went downtown” using Char4Grams event driver on the text, the final event set contains [Mike|ike |ke w|e we| wen|went|ent |nt d|t do| dow|down|ownt|wnto|ntow|town]. This similar approach is also done for Char8Grams, except it extracts the individual characters 8 at a time.

These three comparison event drivers (Words, Char4Grams, and Char8Grams) will be evaluated against WordsBeforeAfterNamedEntities event driver. They were chosen since past research proves these comparison event drivers to be the current standard, state-of-the-art approaches, to computer-based authorship attribution. CharNGrams, in particular, has many scholarly articles written about its performance in authorship attribution. In one of these articles, CharNGrams was tested to attribute authorship in English, Greek and Chinese languages. They tested CharNGrams and found when using this approach as an event driver "the accuracy of the results is at the level of the current state of the art approaches or higher in some cases" (Keselj, Peng, Cercone, & Thomas, 2003).

2.3 Stanford Named Entity Recognizer

Multiple named entity recognizers exist and are produced by individuals, teams and universities. On-going research concerned with the ability of computer recognition and categorization (person, date, organization, etc.) of named entities has been in progress world-wide since 1990. Rather than re-invent the wheel, Stanford's Named Entity Recognizer is introduced and modified in JGAAP to support this research. This modified version became the WordsBeforeAfterNamedEntities event driver.

The Stanford Named Entity Recognizer is a Java implementation of a Named Entity Recognizer. The Stanford Named Entity Recognizer (NER) "labels sequences of words in a text which are the names of things, such as person and company names, or gene and protein names. The software provides a general (arbitrary order) implementation of linear chain Conditional Random Field (CRF) sequence models, coupled with well-engineered feature extractors for Named Entity Recognition" (Finkel, Grenager, & Manning, 2005).

Chapter 3: Materials

The testing was executed on numerous batches of different works composed by multiple authors. The research specifically employed the AAAC corpus built into JGAAP. The AAAC (Ad hoc Authorship Attribution Competition) was designed by Dr. Patrick Juola and is a "moderate-scale empirical test bed for the comparative evaluation of authorship attribution methods" (Juola & Vescovi, 2011). This corpus consists of 13 problems, however only seven of them are in English. For this research, I concentrated only on texts written in the English language. The problems written in English are Problems A, B, C, D, E, G, and H. Within each distinct problem, the samples consisted of a genre (i.e. Romance, Fantasy, Plays, etc.) with texts of similar lengths (short stories, novels).   The problems, as a whole, vary in length and contain multiple authors within each problem. There also existed texts in the problem where there was no known author. The samples where the correct author did not exist in the problem were left out of the analysis and research. This was done since the primary concern was to attribute authorship when the correct author's training documents were mixed in with other authors.

Other built-in functions of JGAAP that were utilized are the event drivers previously discussed. This enabled one to statistically compare results from the WordsBeforeAfterNamedEntities event driver to other event drivers that have been shown to be statistically sound at attributing authorship. Specifically, the research looked at the following event drivers: Words and CharNGrams. For the CharNGrams, the N chosen was 4 and 8. The N denotes how many characters of a word JGAAP will analyze in a sequence.

In addition, out of pure curiosity, two additional event drivers were added to the JGAAP framework. They are WordsOnlyBeforeNamedEntities and WordsOnlyAfterNamedEntities. Looking at these separately may lead to research into another area that will be further explained in the 6.6 Improvements section. They were coded similarly to the WordsBeforeAfterNamedEntities event driver.

R Statistical Software and Microsoft Excel were utilized in order to complete McNemar's test and the Meta-Analysis discussed in the Methods section.

Chapter 4: Methods

4.1 Pure Chance

In order to establish the probability of JGAAP choosing the correct author by chance, the predictive probability was determined first. The following table shows the probability of choosing the correct author for each AAAC problem by chance.

Pure Chance Authorship Attribution

Problem	Number of Samples	Number of Authors	Pure Chance of Choosing Correct Author
A	13	13	0.076923
B	13	13	0.076923
C	9	5	0.200000
D	3	3	0.333333
E	3	3	0.333333
G	4	2	0.500000
H	3	3	0.333333

4.2 Baseline Results

Following this, baseline results were found. These baseline results were performed by executing the JGAAP GUI. JGAAP performed each analyses (WEKA SMO, Centroid Driver: Alternate Intersection Distance, and Centroid Driver: Cosine Distance) on every distinct English problem in the AAAC Corpus and ran for each event driver (Words, Char4Grams, Char8Grams, WordsBeforeAfterNamedEntities, WordsOnlyBeforeNamedEntities and WordsOnlyAfterNamedEntities).

4.2.1 JGAAP Output

For each problem, JGAAP returned the authors in a rank format. An example of one is following.

```
Correct: Author07 /com/jgaap/resources/aaac/problemA/Asample04.txt
Canonicizers:
    none
EventDrivers:
    Words Before and After Named Entities
Analysis:
    WEKA SMO f : false, v : -1, g : 0.01, e : 1, r : Polynomial, c : 1, n : normalize, o : false
1. Author10 0.14285714285714285
2. Author04 0.13186813186813187
3. Author07 0.12087912087912088
4. Author01 0.10989010989010989
5. Author05 0.0989010989010989
6. Author13 0.08791208791208792
7. Author11 0.07692307692307693
8. Author02 0.06593406593406594
9. Author09 0.054945054945054944
10. Author08 0.04395604395604396
11. Author06 0.03296703296703297
12. Author12 0.02197802197802198
13. Author03 0.01098901098901099
```

This shows Problem A, Sample04 results. Notice the correct author of Problem A, Sample04 is Author07. The event driver (feature) used was WordsBeforeAfterNamedEntities and the analysis was WEKA SMO. We notice the correct author was chosen in third place out of the thirteen different authors being compared to sample04. Note, when using the WordsBeforeandAfterNamedEntities event driver with WEKA SMO analysis, the best predicted author for Problem A, Sample04 is Author10.

4.2.2 First Place Comparison and Un-weighted Accuracy Percentages

After receiving the results for each sample and problem, a table was created to sum the number of times an event driver correctly identified the correct author. This was done for each event driver (Words, Char4Grams, Char8Grams, WordsBeforeAfterNamedEntities, WordsOnlyBeforeNamedEntities, and WordsOnlyAfterNamedEntities) across every problem in the AAAC corpus and for each type of analysis (WEKA SMO, Centroid Driver: Alternate Intersection Distance, Centroid Driver: Cosine Distance). These three tableaux are listed in the Results section.

4.2.3 2x2 Contingency Tables

After finding the above for each distinct problem in the AAAC corpus, tables were compiled for first place comparisons. This data was then formatted into 2x2 contingency tables to compare each event driver with the WordsBeforeAfter event driver. The 2x2 contingency tables are listed in the Results section.

4.3 Non-Parametric Statistics

For the AAAC corpus, sample sizes for each problem are not very large. Even when pooled together, the sample size is still under 50 and for most statistical analyses it does not meet many of the underlying statistical assumptions to test. In addition, the population variance and mean are unknown. In order to analyze statistically, we must use methods that are said to be distribution-free. In other words, the methods are based on functions of the sample observations whose corresponding random variables have a distribution that does not depend on the specific distribution function of the population from which the sample was drawn. Because of this, assumptions regarding the underlying population are not necessary (Gibbons & Chakraborti, 2011). This leads us to use two non-parametric statistics. The non-parametric statistics chosen to use were McNemar’s Test and the Sign Test. These both use paired data. Paired data consists of observations in the first group (in this thesis, a comparison event driver) which have a corresponding observation in the second group (the WordsBeforeAfterNamedEntities event driver) (Pagano & Gauvreau, 2000). We can pair this data since each comparison event driver is running on the exact same samples and problems as the WordsBeforeAfterNamedEntities event driver.

4.4 McNemar's Test

McNemar's test is used for dichotomous nominal variables. Here the dichotomy is 1 or 0, where 1 represents success and 0 failure. For this research, 1 represents that the correct author came in first place for the given problem and sample. 0 represents any other rank received. We represent the data in a 2x2 contingency table by summing the number of 1's and 0's each event driver received. Below is an example:

		Char4Grams (using WEKA)		
		C	I	
Words B/A	C	7	7	14
	I	21	13	34
		28	20	48

In this example, we are comparing the Char4Grams event driver against the WordsBeforeAfterNamedEntities event driver using WEKA SMO analysis. The C stands for Correct (successes, 1's), in other words it produced the correct author. I stands for incorrect (failures, 0's), or in other words it did not choose the correct author. In the above example, we can make the following descriptive conclusions, Char4Grams named 28 out of 48 correct authors. WordsBeforeAfterNamedEntities named 14 out of 48 correct authors.

In a 2x2 contingency table, there are two types of pairs, concordant and discordant. Concordant pairs – or the pairs of responses in which both events got the sample correct or incorrect – provide no information for testing a null hypothesis about differences in the two event drivers. On the other hand, the discordant pairs – or the pairs of responses where one event got a sample correct and the other got the same sample incorrect and vice versa, provide the insight we need in order to complete McNemar's test (Pagano & Gauvreau, 2000). In the previous example, the discordant pairs would be 7 (C, I) and 21 (I, C).

The ratio of McNemar's test has a chi-squared distribution with 1 degree of freedom (Gibbons & Chakraborti, 2011). This ratio was calculated using R Statistical Software. The formula that R utilizes is $x^2 = \frac{(b-c-1)^2}{(b+c)}$. The command for this in R Statistical Software is mcnemar.test(*matrixname)*.

The output of McNemar's test is listed in the Results section.

4.5 Sign Test

In addition to McNemar's test for paired data, the Sign Test was used. This is also a non-parametric test. The Sign Test enabled this research to look into the various different rankings a sample could receive. In this test, we run the event drivers Words, Char4Grams, and Char8Grams individually against the WordsBeforeAfter for each method of analysis (WEKA SMO, Centroid Driver: Alt Int Distance, and Centroid Driver: Cosine Distance). This is done by taking the rank of a single sample from using one of the Words, Char4Grams, or Char8Grams and subtracting the same sample rank of the WordsBeforeAfter to get a difference, $D_i = X_i - Y_i$, where Y_i is always the rank for the WordsBeforeAfter Event Driver. For example, if the rank of sample A01 for Char8Grams using the Centroid Driver: Alt Int Distance was 4 (X_i = 4) and the rank of the same sample A01 for WordsBeforeAfter using the Centroid Driver: Alt Int Distance was 1 (Y_i = 1), we find the difference by subtracting the rank of the WordsBeforeAfter from the Char8Gram rank. We then see 4 – 1 = +3. The difference is then +3, so d_i = +3. After finding all the differences for the comparison event drivers and the WordsBeforeAfter event driver, we sum only the positive differences, not by the number, but by the count of "+". For example, using Centroid Driver: Alt Int Distance and comparing the differences between Char8Grams and WordsBeforeAfter, we receive 8 positive differences, 19 with no difference, and 21 negative differences. This was repeated for each comparison driver against the WordsBeforeAfterNamedEntities event driver for each method of JGAAP analysis.

The test statistic for the Sign Test becomes the number of positive differences, denoted as M. The n, or sample size, is the remaining pairs after ignoring the zero (or no) difference ranks. We find the p-value by using the binomial distribution.

The null hypothesis for this test is H_0: p = 0.5. The alternative hypothesis is H_A: p ≠ 0.5. This assumes that the event driver being compared to WordsBeforeAfterNamedEntities will attribute authorship the same way under the same analysis. Therefore, it is tested that p = 0.5, since one would expect the same number of positive differences "+'s" as negative differences "-'s" (Berry & Lindgren, 1996).

Using R Statistical software, the binomial probability is computed, so that x = M (where M is the test statistic) given some n (where n is the sample size), $P(x = M|N = n)$. The outcome of the tests are shown in tableaux format in the Results section.

4.6 Meta-Analysis

This research looked at the summation of all of the problems in the AAAC corpus and weighted them according to the various numbers of authors and samples in the problem. This allowed one to examine how the various event drivers worked over a group of diverse genres and lengths of texts. The research then encompassed the ability to make global comparisons between each specific event driver. This was done using meta-analytical techniques.

In the AAAC corpus, each problem consisted of a certain number of samples. The number of samples in each problem varied. In the prior statistical analyses, this did not matter due to the nature of the non-parametric statistical assumptions. For this analysis method, this does matter. In order to resolve this issue, weights must be in place. Meta-analysis refers to a technique of assessing data that essentially combines results of other studies (Triola & Triola, 2006). For this research, one can sum the results for each problem (A, B, C, D, E, G, and H), into one large study using meta-analytical techniques.

Meta-analysis represents each problem's findings in the form of effect sizes. An effect size is a statistic that encodes the essential quantitative information from each problem. This effect size statistic is based on the theory of standardization. The most common effect size statistics in meta-analysis standardize on the variation in the sample distributions of scores for the measures of interest (Lipsey & Wilson, 2001). In this research, standardization occurs on the variation within each problem in the AAAC Corpus.

A meta-analysis can only be completed when the smaller studies that encompass the large meta-analysis are identical. Not only do they need to have the same dependent and independent variables, but they also must be using the same statistical measures and analysis in order to combine them into a larger problem.

For this research, the meta-analysis is done three times, one for each JGAAP analysis (WEKA SMO, Centroid Driver: Alternate Intersection Distance, and Centroid Driver: Cosine Distance).

First, the baseline accuracy must be determined for each event driver individually on every distinct AAAC problem. This yields the $\hat{p}_i$ values which are acquired by dividing the number correct in each respective AAAC problem by the number of samples (n) in the respective problem. Using this information, the corresponding variances are obtained for each single problem given by the formula $\sigma^2 = \frac{\hat{p}\hat{q}}{n}$. The weight then becomes the reciprocal of the variance or $w_i = \frac{1}{\sigma^2}$. We then find the weighted average accuracy for an event driver by taking the sum of each AAAC problem weight, w_i multiplied by the respective AAAC problem $\hat{p}_i$.

Now that there are weighted average accuracies, one can find the corresponding weighted standard error and the weighted 95% confidence intervals. The weighted 95% confidence intervals of each comparison event driver will be contrasted to the WordsBeforeAfterNamedEntities event driver.

The results of the meta-analysis are listed in the Results section.

Chapter 5: Results

The results are shown in tableau format.

5.1 Baseline Results

5.1.1 JGAAP Output

The following tableaux list out the rank of the correct author for each single sample. They also provide totals for how many correct were found as well as the average rank. The problems are listed in alphabetical order of the AAAC corpus. Three tableaux exist for each problem in the AAAC corpus: the first is the WEKA SMO analysis, the second is Centroid Driver: Alt Int Distance and the third is the Centroid Driver: Cosine Distance. An asterisk (*) next to the number indicates a tie in the first place rank.

Problem A using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	12	12	11	13	13	13
02	2	1	4	11	11	6
03	3	3	4	2	6	8
04	1	1	4	3	1	7
05	2	1	2	1	2	1*
06	10	4	9	1	1	10
07	3	1	3	10	5	9
08	1	1	1	10	8	8
09	1	1	1	10	9	4
10	1	1	3	3	2	3
11	5	5	8	10	7	12
12	8	8	13	9	6	10
13	1	1	3	6	2	11
TOTAL	5	8	2	2	2	1*
Average Rank	3.84615	3.07692	5.07692	6.84615	5.61538	7.84615

Problem A using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	2	2	4	1	2	3
02	1	1	1	7	3	1*
03	4	5	1	10	1*	9
04	1	1	1	9	1*	5
05	2	4	1	1*	1*	1*
06	1	1	1	3	1	4
07	1	6	2	10	6	3
08	5	8	2	1*	1*	9
09	1	5	1	1*	1*	1*
10	1	4	1	1*	1*	1*
11	4	4	3	7	5	1*
12	3	11	5	1*	13	13
13	1	6	1	10	1*	12
TOTAL	7	3	8	6*	8*	5*
Average Rank	2.07692	4.46154	1.84615	4.76923	2.84615	4.84615

Problem A using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	5	7	4	7	2	3
02	1	10	1	6	12	7
03	3	1	1	4	2	7
04	1	1	1	13	2	11
05	6	2	1	1	4	8
06	4	2	1	5	2	3
07	5	2	1	12	2	12
08	2	6	3	9	11	9
09	8	3	1	6	6	6
10	4	1	1	6	6	6
11	1	3	6	10	9	4
12	1	1	1	7	1	13
13	7	5	2	9	1	11
TOTAL	4	4	9	1	2	0
Average Rank	3.69231	3.38462	1.84615	7.30769	4.61538	7.69231

Problem B using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	2	4	1	1	6	1
02	1	5	4	11	9	11
03	1	1	3	4	9	3
04	1	1	1	1	1	2
05	4	1	6	4	2	5
06	10	9	10	13	13	13
07	6	1	8	3	1*	3
08	2	2	4	7	4	9
09	3	1	3	7	9	7
10	10	1	10	12	10	11
11	13	13	13	6	8	9
12	10	11	8	9	4	12
13	1	2	1	3	1	7
TOTAL	4	6	3	2	3*	1
Average Rank	4.92308	4.00000	4.61538	6.23077	5.92308	7.15385

Problem B using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	6	10	5	5	6	6
02	5	10	5	7	8	6
03	1	1	1	6	3	6
04	3	9	7	2	1*	5
05	2	6	2	9	9	8
06	1	1	1	1*	3	1
07	1	4	1	10	12	3
08	3	5	1	2	3	3
09	4	4	1	5	7	3
10	1	1	1	1	1*	2
11	12	12	11	13	13	13
12	4	4	3	4	3	3
13	1*	8	1	9	8	11
TOTAL	5*	3	7	2*	2*	1
Average Rank	3.38462	5.76923	3.07692	5.69231	5.92308	5.38462

Problem B using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	11	10	6	10	12	1
02	12	10	8	11	11	12
03	2	11	1	2	1	4
04	7	8	3	3	3	1
05	1	7	7	9	9	9
06	4	4	7	1	3	1
07	8	6	4	8	6	9
08	4	2	2	7	9	6
09	1	1	1	2	2	3
10	2	1	1	1	1	3
11	9	7	10	13	12	13
12	8	5	10	6	6	10
13	4	4	2	2	2	7
TOTAL	2	2	3	2	2	3
Average Rank	5.61538	5.84615	4.76923	5.76923	5.92308	6.07692

Problem C using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	4	2	4	1	1	2
02	1	1	1	1	1	1
03	1	2	1	1	1	2
04	1	1	1	4	3	3
05	4	1	5	4	5	2
06	1	1	1	4	4	1
07	1	1	1	2	2	2
08	5	4	5	5	5	4
09	1	1	1	1	1	1
TOTAL	6	6	6	4	4	3
Average Rank	2.11111	1.55556	2.22222	2.55556	2.55556	2.00000

Problem C using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	2	2	3	3	4	3
02	1	1	1	1	1	1
03	2	2	2	1	1	1
04	1	1	1	2	2	2
05	3	5	4	5	5	5
06	1	1	1	2	2	3
07	1	1	1	1*	1	2
08	4	5	5	4	4	5
09	1	1	1	1	1	2
TOTAL	5	5	5	4*	4	2
Average Rank	1.77778	2.11111	2.11111	2.22222	2.33333	2.66667

Problem C using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	1	1	2
02	1	1	1	1	4	1
03	3	1	1	2	2	1
04	1	1	1	3	4	2
05	1	1	1	2	1	4
06	1	1	1	1	1	1
07	1	3	2	2	3	1
08	3	3	4	1	1	4
09	2	1	1	1	1	1
TOTAL	6	7	7	5	5	5
Average Rank	1.55556	1.44444	1.44444	1.555556	2.00000	1.88889

Problem D using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	1	1	1
02	3	2	3	2	1	3
04	1	1	1	1	1	1
TOTAL	2	2	2	2	3	2
Average Rank	1.66667	1.3333	1.66667	1.3333	1.0000	1.66667

Problem D using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	1	1	1
02	3	2	3	2	1	3
04	1	1	1	1	1	1
TOTAL	2	2	2	2	3	2
Average Rank	1.66667	1.33333	1.66667	1.33333	1.00000	1.66667

Problem D using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	1	1	1
02	2	1	1	1	2	1
04	1	1	1	3	3	2
TOTAL	2	3	3	2	1	2
Average Rank	1.33333	1.00000	1.00000	1.66667	2.00000	1.33333

Problem E using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	1	1	2
02	1	1	2	1	2	1
04	3	3	3	3	3	2
TOTAL	2	2	1	2	1	1
Average Rank	1.66667	1.66667	2.00000	1.66667	2.00000	1.66667

Problem E using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	2	1	1	1	1	1
02	1	1	1	2	2	2
04	3	3	3	3	2	3
TOTAL	1	2	2	1	1	1
Average Rank	2.00000	1.66667	1.66667	2.00000	1.66667	2.00000

Problem E using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	3	3	1
02	3	3	3	1	1	2
04	1	1	1	1	1	1
TOTAL	2	2	2	2	2	2
Average Rank	1.66667	1.66667	1.66667	1.66667	1.66667	1.33333

Problem G using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	2	2	2	2
02	1	1	1	2	2	2
03	1	2	2	1	1	1
04	2	1	1	1	1	2
TOTAL	3	3	2	2	2	1
Average Rank	1.25000	1.25000	1.50000	1.50000	1.50000	1.75000

Problem G using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	2	2	2
02	2	2	2	1	1	1
03	1	1	1	1	1	1
04	2	2	2	2	2	2
TOTAL	2	2	2	2	2	2
Average Rank	1.50000	1.50000	1.50000	1.50000	1.50000	1.50000

Problem G using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	1	1	2	1	2
02	2	2	2	1	1	1
03	1	1	1	1	1	1
04	1	1	1	1	2	1
TOTAL	3	3	3	3	3	2
Average Rank	1.25000	1.25000	1.25000	1.25000	1.25000	1.25000

Problem H using WEKA SMO Analysis

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	1	3	1	2	2	1
02	2	2	2	3	3	2
03	2	1	3	3	3	3
TOTAL	1	1	1	0	0	1
Average Rank	1.66667	2.00000	2.00000	2.66667	2.66667	2.00000

Problem H using Centroid Driver: Alt Int Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	3	3	3	3	3	2
02	2	2	2	2	2	3
03	1	1	1	1	1	1
TOTAL	1	1	1	1	1	1
Average Rank	2.0000	2.00000	2.00000	2.000000	2.00000	2.00000

Problem H using Centroid Driver: Cosine Distance

Sample	Words	Char4Gram	Char8Gram	WordsBef/Aft	WordsBef	WordsAft
01	3	3	3	2	2	2
02	2	2	2	3	3	1
03	1	1	1	1	1	1
TOTAL	1	1	1	1	1	2
Average Rank	2.00000	2.00000	2.00000	2.000000	2.00000	1.33333

5.1.2 First Place Comparison and Un-weighted Accuracy Percentages

The following three tables illustrate the number of times an event driver was able to identify the correct author. The first table is for the WEKA SMO analysis, the second is the Centroid Driver: Alternate Intersection Distance, and the third is the Centroid Driver: Cosine Distance. Each table shows the summation of correct authorship for all six event drivers per analysis. In addition, it gives an un-weighted accuracy percentage. An asterisk (*) next to the number indicates a tie in the first place rank.

WEKA SMO Analysis

AAAC	N	Words	Char4Gram	Char8Gram	WordsBef/Aft	Words Before	Words After
A	13	5	8	2	2	2	1*
B	13	4	6	3	2	3*	1
C	9	6	6	6	4	4	3
D	3	2	2	2	2	3	2
E	3	2	2	1	2	1	1
G	4	3	3	2	2	2	1
H	3	1	1	1	0	0	1
TOTAL	48	23	28	17	14	15	10
% ACC		0.4792	0.5833	0.3542	0.2917	0.3125	0.2083

Centroid Driver: Alt Int Distance Analysis

AAAC	N	Words	Char4Gram	Char8Gram	WordsBef/Aft	Words Before	Words After
A	13	7	3	8	6*	8*	5*
B	13	5*	3	7	2*	2*	1
C	9	5	5	5	4*	4	2
D	3	2	2	2	2	3	2
E	3	1	2	2	1	1	1
G	4	2	2	2	2	2	2
H	3	1	1	1	1	1	1
TOTAL	48	23	18	27	18	21	14
%ACC		0.4792	0.3750	0.5625	0.3750	0.4375	0.2917

Centroid Driver: Cosine Distance Analysis

AAAC	N	Words	Char4Gram	Char8Gram	WordsBef/Aft	Words Before	Words After
A	13	4	4	9	1	2	0
B	13	2	2	3	2	2	3
C	9	6	7	7	5	5	5
D	3	2	3	3	2	1	2
E	3	2	2	2	2	2	2
G	4	3	3	3	3	3	2
H	3	1	1	1	1	1	2
TOTAL	48	20	22	28	16	16	19
%ACC		0.4167	0.4583	0.5833	0.33333	0.3333	0.3958

5.1.3 Contingency Tables

After receiving the above results, 2x2 contingency tableaux were constructed. These tableaux represent a comparison event driver (Words, Char4Grams, Char8Grams) being compared to the WordsBeforeAfterNamedEntities event driver. Again, we show these tables for each different JGAAP analysis. The first shows the WEKA SMO analysis, the second is Centroid Driver: Alt Int Distance, and the third is the Centroid Driver: Cosine Distance. The following are the 2x2 contingency tables that are summed over the whole AAAC Corpus.

5.1.4 WEKA SMO 2x2 Contingency Tables

WEKA SMO		Words		
		C	I	
Words B/A	C	9	5	14
	I	14	20	34
		23	25	48

WEKA SMO		Char4Grams		
		C	I	
Words B/A	C	7	7	14
	I	21	13	34
		28	20	48

WEKA SMO		Char8Grams		
		C	I	
Words B/A	C	9	5	14
	I	8	26	34
		17	31	48

5.1.2 Centroid Driver: Alternate Intersection Distance 2x2 Contingency Tables

ALT INT		Words		
		C	I	
Words	C	11	7	18
B/A	I	12	18	30
		23	25	48

ALT INT		Char4Grams		
		C	I	
Words	C	10	8	18
B/A	I	8	22	30
		18	30	48

ALT INT		Char8Grams		
		C	I	
Words	C	13	5	18
B/A	I	14	16	30
		27	21	48

5.1.3 Centroid Driver: Cosine Distance 2x2 Contingency Tables

COSINE		Words		
		C	I	
Words	C	8	8	16
B/A	I	12	20	32
		20	28	48

COSINE		Char4Grams		
		C	I	
Words	C	10	6	16
B/A	I	12	20	32
		22	26	48

COSINE		Char8Grams		
		C	I	
Words	C	12	4	16
B/A	I	16	16	32
		28	20	48

5.2 McNemar's Test

From these 2x2 contingency tables, McNemar's test was applied. These tests were completed using R Statistical Software. The results are listed below using the overall AAAC corpus. All problems are summed together due to the non-parametric nature of McNemar's test to make for a total of 48 different samples to test. Both the x^2 test statistic and p-value are listed. The first table is the WEKA SMO analysis, the second is the Centroid Driver: Alternate Intersection Distance and the third is the Centroid Driver: Cosine Distance. The italicized x^2 and p-values are significant.

5.2.1 McNemar's Test using WEKA SMO analysis

WEKA SMO	x^2	p-value
Words and WordsBeforeAfterNamedEntities	3.3684	0.0665
Char4Grams and WordsBeforeAfterNamedEntities	*6.0357*	*0.0140*
Char8Grams and WordsBeforeAfterNamedEntities	0.3077	0.5791

5.2.2 McNemar's Test using Centroid Driver: Alternate Intersection Distance analysis

Centroid Driver: Alt Intersection Distance	x^2	p-value
Words and WordsBeforeAfterNamedEntities	0.8421	0.3588
Char4Grams and WordsBeforeAfterNamedEntities	0.0000	1.0000
Char8Grams and WordsBeforeAfterNamedEntities	3.3684	0.06646

5.2.3 McNemar’s Test using Centroid Driver: Cosine Distance analysis

Centroid Driver: Cosine Distance	x^2	p-value
Words and WordsBeforeAfterNamedEntities	0.4500	0.5023
Char4Grams and WordsBeforeAfterNamedEntities	1.3889	0.2386
Char8Grams and WordsBeforeAfterNamedEntities	*6.0500*	*0.0139*

McNemar’s Test was used in this research for when two different event drivers disagree on a given sample. It lets one know whether or not the event drivers are different based on the discordant pairs. McNemar’s Test specifically looks at when an error occurs (in this case if the correct author was not found), are the event drivers evenly split on the disagreements. In addition, this test is not used for accuracy but rather for error or spurious analysis in the results.

Since McNemar’s test was used to compare Words to WordsBeforeAfterNamedEntities, Char4Grams to WordsBeforeAfterNamedEntities, and Char8Grams to WordsBeforeAfterNamedEntities, the p-value must be adjusted to consider the three different multiple comparisons of event drivers. The p-value was originally established as $p =$ 0.05. To adjust for the three multiple comparisons, the p-value will be divided by 3. Therefore, the *p-value* = 0.01667. In order to reject the null hypothesis, the p-value that corresponds to the chi-squared approximation must be smaller than p = 0.01667.

From the above results, Char4Grams was significantly better at correctly identifying authors than WordsBeforeAfterNamedEntities when using the WEKA SMO analysis since the chi-squared test statistic was 6.0357 with a corresponding p-value of 0.0140.

In addition, Char8Grams was significantly better at correctly identifying the correct author of a sample than WordsBeforeAfterNamedEntities when using the Centroid Driver: Cosine Distance analysis. One can see from the aforementioned results that the chi-squared test statistic was 6.0500 with a corresponding p-value of 0.0139.

5.3 Sign Test

The Sign Test was also employed to analyze results. The null hypothesis for this test is H_0: p = 0.5. The alternative hypothesis is

H_A: p ≠ 0.5. This assumes that the event driver being compared to WordsBeforeAfterNamedEntities will attribute authorship the same way under the same JGAAP analysis. These tableaux are listed below:

5.3.1 Sign Test using Words and WordsBeforeAfterNamedEntities

Words/WordsBeforeAfterNamedEntities

Analysis Method	M (total # positive difference	N (pairs remaining after ignoring zero)	P(x ≤M | n = N)	Two Sided P-value
WEKA SMO	10	35	0.008336924	0.01667385
Centroid: Alt Int	11	32	0.0550928	0.1101842
Centroid: Cosine	15	38	0.1279375	0.2258751

5.3.2 Sign Test using Char4Grams and WordsBeforeAfterNamedEntities

Char4Grams/WordsBeforeAfterNamedEntities

Analysis Method	M (total # positive difference	N (pairs remaining after ignoring zero)	P(x ≤ M | n = N)	Two Sided P-value
WEKA SMO	9	37	0.001281604	0.002563208
Centroid: Alt Int	13	31	0.2365648	0.4731297
Centroid: Cosine	12	36	0.03262267	0.06524534

5.3.3 Sign Test using Char8Grams and WordsBeforeAfterNamedEntities

Char8Grams/WordsBeforeAfterNamedEntities

Analysis Method	M (total # positive difference	N (pairs remaining after ignoring zero)	P(x ≤ M | n = N)	Two Sided P-value
WEKA SMO	13	34	0.1147405	0.229481
Centroid: Alt Int	8	29	0.01205977	0.02411954
Centroid: Cosine	6	33	0.0001620317	0.0003240635

The Sign Test considered the different rankings possible for each sample. This is done by obtaining the difference between the rank of a given sample using a comparison event driver to the WordsBeforeAfterNamedEntities event driver. The difference in the rankings will not be a continuous variable, it will instead be either "+" or "-". Since the Sign Test compares Words to WordsBeforeAfterNamedEntities, Char4Grams to WordsBeforeAfterNamedEntities, and Char8Grams to WordsBeforeAfterNamedEntities, one must adjust the p-value for the three multiple comparisons. The p-value was originally established as $p =$ 0.05. To adjust for the three comparisons, the p-value is divided by 3. Therefore, the *p-value* = 0.01667. In order to reject the null hypothesis that we are equally likely to receive "+" as well as a "-" for each event driver, we must find that the p-value for the Sign Test is less than $p = 0.01667$.

From the above results, one can say with statistical significance that the Words event driver out performs the WordsBeforeAfterNamedEntities event driver using WEKA SMO analysis in JGAAP. Also when using WEKA SMO analysis, the Char4Grams event driver correctly attributes authorship more frequently than the WordsBeforeAfterNamedEntities event driver. When using Centroid Driver: Cosine

Distance, the Char8Grams event driver significantly outperforms the WordsBeforeAfterNamedEntities event driver. This last one is the most significant result with a p-value of 0.000324.

When using the Sign Test there was no difference between the comparison event drivers against the WordsBeforeAfterNamedEntities event driver when using the Centroid Driver: Alt Int Distance analysis in JGAAP.

5.4 Meta-Analysis

Meta-analysis is a statistic that allows one to sum all of the problems in the AAAC corpus and weight them accordingly. The following are the results. They are listed by JGAAP analysis measures. The first is the WEKA SMO, the second is the Centroid Driver: Alt Int Distance and the third is the Centroid Driver: Cosine Distance.

Meta-Analysis: Words with Centroid Driver: WEKA SMO

	Un-weighted	Weighted
Accuracy	0.479167	0.482865
Standard Error	0.72104	0.067684
Lower 95% Confidence Interval	0.337843	0.350204
Upper 95% Confidence Interval	0.620491	0.615526

Meta-Analysis: Char4Grams with Centroid Driver: WEKA SMO

	Un-weighted	Weighted
Accuracy	0.583333	0.589035
Standard Error	0.0711620	0.069078
Lower 95% Confidence Interval	0.448520	0.453642
Upper 95% Confidence Interval	0.722808	0.724428

Meta-Analysis: Char8Grams with Centroid Driver: WEKA SMO

	Un-weighted	Weighted
Accuracy	0.3541667	0.315678
Standard Error	0.069029	0.060847
Lower 95% Confidence Interval	0.21887	0.196418
Upper 95% Confidence Interval	0.489464	0.434938

Meta-Analysis: WordsBeforeAfter with Centroid Driver: WEKA SMO

	Un-weighted	Weighted
Accuracy	0.3020833	0.254432
Standard Error	0.066272	0.05766
Lower 95% Confidence Interval	0.17219	0.141418
Upper 95% Confidence Interval	0.431976	0.367446

Meta-Analysis: Words with Centroid Driver: ALT INT Analysis

	Un-weighted	Weighted
Accuracy	0.4791667	0.477268
Standard Error	0.072104	0.070678
Lower 95% Confidence Interval	0.337843	0.338739
Upper 95% Confidence Interval	0.620491	0.615797

Meta-Analysis: Char4Grams with Centroid Driver: ALT INT Analysis

	Un-weighted	Weighted
Accuracy	0.520833	0.517856
Standard Error	0.072104	0.064628
Lower 95% Confidence Interval	0.379509	0.391185
Upper 95% Confidence Interval	0.662157	0.644527

Meta-Analysis: Char8Grams with Centroid Driver: ALT INT Analysis

	Un-weighted	Weighted
Accuracy	0.5208333	0.52108
Standard Error	0.072104	0.070221
Lower 95% Confidence Interval	0.379509	0.383447
Upper 95% Confidence Interval	0.662157	0.658713

Meta-Analysis: WordsBeforeAfter with Centroid Driver: ALT INT Analysis

	Un-weighted	Weighted
Accuracy	0.375	0.333328
Standard Error	0.069878	0.063872
Lower 95% Confidence Interval	0.238038	0.208139
Upper 95% Confidence Interval	0.511962	0.458517

Meta-Analysis: Words with Centroid Driver: Cosine Analysis

	Un-weighted	Weighted
Accuracy	0.4375	0.396655
Standard Error	0.071603	0.062365
Lower 95% Confidence Interval	0.297158	0.27442
Upper 95% Confidence Interval	0.577842	0.51889

Meta-Analysis: Char4Grams with Centroid Driver: Cosine Analysis

	Un-weighted	Weighted
Accuracy	0.515463	0.4909292
Standard Error	0.072132	0.0596378
Lower 95% Confidence Interval	0.374085	0.374039
Upper 95% Confidence Interval	0.656843	0.607819

Meta-Analysis: Char8Grams with Centroid Driver: Cosine Analysis

	Un-weighted	Weighted
Accuracy	0.4742268	0.4945039
Standard Error	0.072069	0.0616334
Lower 95% Confidence Interval	0.332972	0.373702
Upper 95% Confidence Interval	0.615482	0.615305

Meta-Analysis: WordsBeforeAfter with Centroid Driver: Cosine Analysis

	Un-weighted	Weighted
Accuracy	0.3333333	0.231348
Standard Error	0.068044	0.051216
Lower 95% Confidence Interval	0.199967	0.130965
Upper 95% Confidence Interval	0.466699	0.331731

Using meta-analytical techniques enabled one to weight the data according to a problems' sample size. After weighting the average, variance and standard error, the weighted 95% confidence intervals were constructed above.

In the WEKA SMO analysis, the weighted 95% confidence intervals for Words and Char4Grams are completely higher than the weighted 95% confidence interval for WordsBeforeAfterNamedEntities. This statistically ascertains that the Words and Char4Grams event drivers surpass the WordsBeforeAfterNamedEntities event driver when attributing authorship correctly. In addition, when using the WEKA SMO analysis, the weighted 95% confidence intervals for Char8Grams and WordsBeforeAfterNamedEntities had plenty of overlap. Therefore, when using the WEKA SMO analysis, one cannot statistically say Char8Grams performs any differently than WordsBeforeAfterNamedEntities event driver at predicting authorship.

In the Centroid Driver: Alt Int Distance analysis, the weighted 95% confidence intervals do not provide any statistically significant data to present that one event driver works better than another.

In the Centroid Driver: Cosine Distance analysis, the weighted 95% confidence intervals for Char4Grams and Char8Grams are completely higher than the weighted 95% confidence interval for WordsBeforeAfterNamedEntities. This statistically ascertains that the Char4Grams and Char8Grams event drivers surpass the WordsBeforeAfterNamedEntities event driver when attributing authorship. There is some overlap in the 95% confidence intervals between Words and WordsBeforeAfterNamedEntities.

Chapter 6: Discussion

6.1 Statistical Summary

In summarizing all of the statistical tests, one can see consistent results between the three tests (McNemar's Test, Sign Test, and Meta-Analysis).

When using the WEKA SMO analysis, all three tests showed the Char4Grams event driver consistently choosing the correct author of a document more frequently than the WordsBeforeAfterNamedEntities event driver. In two of the tests, Sign Test and Meta-Analysis, the Words event driver consistently chose the correct author of a document more frequently than the WordsBeforeAfterNamedEntities event driver. When using the WEKA SMO analysis for authorship attribution, WordsBeforeAfterNamedEntities does not attribute authorship as well as the current standard computer-based approaches.

In using the Centroid Driver: Alternate Intersection Distance analysis in JGAAP, there was no significant difference between any of the comparison event drivers and the WordsBeforeAfterNamedEntities event driver. Therefore, the WordsBeforeAfterNamedEntities event driver is not statistically different at predicting authorship of a document than any of the current standard computer-based approaches.

When using the Centroid Driver: Cosine Distance analysis in JGAAP, the three tests ((McNemar's Test, Sign Test, and Meta-Analysis) showed consistent results. The Char8Grams event driver outperformed the WordsBeforeAfterNamedEntities event driver at authorship attribution of a given document. Hence, we can state the Char8Grams event driver predicts an author of a document at a higher percentage rate than the WordsBeforeAfterNamedEntities event driver. The Centroid Driver: Cosine

Distance analysis also showed that Char4Grams was significantly different than the WordsBeforeAfterNamedEntities at predicting authorship in the Meta-Analysis and was very close to significance in the Sign Test.

Consequently, one can conclude that when using WEKA SMO and Centroid Driver: Cosine Distance, the WordsBeforeAfterNamedEntities event driver does not attribute authorship at the current standard of authorship attribution for event drivers. There are other event drivers that exist to predict authorship more correctly than the WordsBeforeAfterNamedEntities event driver.

6.2 Improvements

After completing this research, one can make conclusions regarding the steps and analysis to better improve research in this area.

Looking at function words in particular is a proven approach to attribute authorship. In further examining sentence structure, these function words may not have been directly before or after named entities. To continue we may want to focus on looking either directly before or directly after. One may also want to work at two consecutive words before and after. Because of this idea, a WordsOnlyBeforeNamedEntities and a WordsOnlyAfterNamedEntities event drivers were created during early phases of research. These are coded very similarly to the WordsBeforeAfterNamedEntities; however it only looks to one side of the named entity in a text. Experimental results were not done on these two event drivers. In preliminary analysis, both event drivers showed potential in the baseline results under certain JGAAP analytic measures.

In JGAAP there are 90 different analysis methods. These analysis methods range from Markov Chain analysis to WEKA SMO. It could be possible a different analysis method could enable the WordsBeforeAfterNamedEntities event driver to give better authorship accuracy. Again, the chosen analysis methods for this research were previously statistically proven to aide in authorship attribution.

Several Canonicizers exist in the JGAAP framework. Canonicizers "involve the pre-processing of documents in order to remove unwanted artifacts from those documents" (EVL Labs, 2010). Examples of canonicizers are the stripping a document of white space or punctuation. No need for canoniciziation occurred in order to find the words surrounding named entities so canonicization for this research did not occur. However, one may make a change to this for future considerations.

The AAAC corpus is a corpus that exists specifically to test authorship attribution. Creating a specific corpus to test documents that look explicitly at a certain genre or document length may enable the WordsBeforeAfterNamedEntities event driver to attribute authorship at a higher frequency. Specifically looking at document length, one may choose to look at shorter, untraditional documents such as emails or text messages.

Chapter 7: Conclusion

Authors present their works in various styles of writing. The writing style of an author, formally known as an author's stylometry, can be analyzed through a process known as authorship attribution. Through authorship attribution, one can quantitatively measure frequencies of words and identify probable authorship of a document. Research continues to grow on new techniques to enhance the quality of authorship attribution.

Looking specifically at sentence structure, one notices the frequency of function words found before and after named entities. This research investigated the frequencies of the words surrounding named entities. In statistically evaluating these words, the current best known practices of authorship attribution outperform the WordsBeforeAfterNamedEntities event driver in WEKA SMO and Centroid Driver: Cosine Distance Analysis. However, in Centroid Driver: Alternate Intersection Distance the WordsBeforeAfterNamedEntities event driver is not significantly different at predicting authorship of a document as the current standard event drivers.

This research entailed a new approach for authorship attribution by creating a new event driver that extracts words before and after named entities. This process was completed using JGAAP and implementing the Stanford Named Entity Recognizer. Further research can be done in the area of named entity recognition for authorship attribution of unknown, disputed and forged texts.

REFERENCES

Berry, D., & Lindgren, B. (1996). *Statistics: Theory and methods*. New York , NY: Duxbury Press.

EVL Labs. (2010, November 10). *Documentation*. Retrieved from http://evllabs.com/jgaap/w/index.php/Words

Finkel, J., Grenager, T., & Manning, C. (2005). *Stanford named entity recognizer (ner)*. Retrieved from The Stanford Natural Language Processing Group: http://nlp.stanford.edu/software/CRF-NER.shtml

Frank, E., Legg, S., & Inglis, S. (n.d.). *Class SMO*. Retrieved from WEKA Docs: http://weka.sourceforge.net/doc/weka/classifiers/functions/SMO.html

Gibbons, J., & Chakraborti, S. (2011). *Nonparametric statistical inference*. Boca Raton: Chapman & Hall/CRC.

Juola, P., & Vescovi, D. (2011). Analyzing stylometric approaches to author obfuscation. In G. Peterson, & S. Shenoi, *Advancing Digital Forensics VII* (pp. 115). New York: Springer.

Keselj, V., Peng, F., Cercone, N., & Thomas, C. (2003). N-Gram-Based Author Profiles for Authorship Attribution. *Pacific Association for Computational Linguistics, 9*, 255-264.

Lipsey, M. W., & Wilson, D. B. (2001). *Practical Meta-Analysis*. London: Sage Publications.

Mosteller, F., & Wallace, D. (1963). Inference in an authorship problem. *Journal of the American Statistical Association, 58*(302), 275-309.

Pagano, M., & Gauvreau, K. (2000). *Principles of biostatistics*. Pacific Grove: Brooks/Cole.