Authorship Attribution via Evolutionary Hybridization of Sentiment Analysis, LIWC, and Topic Model Features

by

Josh Gaston

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
December 15, 2018

Keywords: Authorship Attribution, Sentiment Analysis, Topic Model, LIWC, Feature Fusion, Feature Selection

Approved by

Gerry Dozier, Charles D. McCrary Eminent Chair Professor of Computer Science and Software Engineering
Kai Chang, Professor of Computer Science and Software Engineering
Bo Liu, Assistant Professor of Computer Science and Software Engineering

Abstract

Authorship Attribution is a well-studied topic with deep roots in the field of Stylometry. This thesis examines three less traditional feature sets for the purpose of Authorship Attribution. Each feature set is examined alone as well as in combination with the other features. We examine the performance of features derived from Sentiment Analysis, LIWC (Linguistic Inquiry and Word Count), and Topic Models. Using methods from Multimodal Machine Learning, these feature sets are combined in an effort to improve the performance of Authorship Attribution systems. Then a feature selection method based on a Steady-State Genetic algorithm known as GEFeS (Genetic and Evolutionary Feature Selection) is used examine many different subsets of the total feature sets and further improve the performance of the Authorship Attribution Systems.

Acknowledgments

I would like to thank my advisor, Dr. Gerry Dozier, for all his help and guidance through this journey. I would like to thank my other committee members, Dr. Bo Liu and Dr. Kai Chang, for all that they have taught me through my career at Auburn. I would like to thank my friends for the fun times and the good memories and my family for their love and support. To my mom who always motivates me to be better and my dad who never stopped believing in me, I am eternally grateful.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the increased usage of the internet comes the increase in electronic texts found online. These texts appear in many forms such as emails, blogs, or online forums. One problem with text found online is that determining the author can be a difficult task. Documents found online could easily be presented anonymously or even worse presented as having been written by a different author altogether. The ability to determine the true author of a given text is becoming increasingly more useful.

1.1 Authorship Attribution

This thesis focuses on a process known as, Authorship attribution [1, 2, 3, 4], which is a process used to determine the true author of an unknown text. An Authorship Attribution system focuses on characterizing writing styles of a number of authors in an effort to determine which is the author of an unknown text by looking for these defining characteristics. Traditionally, Stylometry Analysis [1, 3, 4] is the preferred method for defining an author's style. Measuring stylometric traits, such as the occurrence of words or characters, of a set of documents with known authors can create quantifiable representations of these documents which would hold information of these authors' styles. These documents could then be compared to documents with unknown authors to determine the most probable author of the unknown text.

1.2 Non-Traditional Feature Sets

A large focus of this thesis is on the representation of the documents. A representation is comprised of a set of features extracted from a document. Researchers have determined a few main categories of features such as lexical, syntactic, semantic, structural, domain-specific, and additional features [3, 4, 5]. These categories are ordered based on their difficulty of extraction from text. Lexical features, such as the counts of word or character occurrences, are the easiest to extract. Other features, like Topic Models and readability metrics which appear in the additional feature category, are more difficult to extract because they require more sophisticated natural language processing techniques. The current state-of-the-art representation is the Bag of Words representation [3] where in a group of documents is represented as a word frequency vector that is comprised of all the words found in the set of documents. Being in the lexical category, not only is it one of the better performing representations it is also one of the easiest to extract from a set of data. This thesis investigates a group of less traditionally used representations. Representations derived from Sentiment Analysis [5, 6, 13, 17, 27, 28], LIWC (Linguistic Inquiry and Word Count) [7, 27, 28], and Topic Models [25, 26, 28] are all examined in this thesis. Subsets, combinations, and combinations of these subsets are examined as well. A feature selection method known as GEFeS (Genetic and Evolutionary Feature Selection) [8, 9, 10] allows many different subsets of each of the feature sets to examined in order to find the most salient features. Concepts from Multimodal Machine Learning [11, 12] are used to fuse the different feature sets together in order to make use of complementary features between feature sets. The combination of these feature selection and feature fusion can be used to not only decrease the size of the feature space but also boost the performance of Authorship Attribution systems. Each feature set is described in more depth in their respective Chapters.

1.3 GEFeS

The feature selection method known as GEFeS uses a Steady-State Genetic Algorithm [9, 10] to evolve a population feature masks. These GEFeS feature masks act as a feature selection method by turning off and on certain features. The chosen features that are turned on and off are evolved based on a fitness function. The function used to determine the fitness of the feature masks took the form of a value preference structure [8] and is based on not only the accuracy obtained by the mask but also the amount of features used. This multi-objective fitness function is designed such that a better fitness is assigned to feature masks that produce a high accuracy and use small percent of the total features. The removal of features not only reduces the computational power needed to classify each instance but also increases the accuracy of the Authorship Attribution system by removing noisy features and leaving only the most salient features behind. More concrete details on the settings used in GEFeS are described in Chapter 2.

1.4 Multimodal Machine Learning

Multimodal Machine Learning is a field of Machine Learning that deals with data that can be captured in different representations or modalities. Traditionally in multimodal machine learning each representation would be associated with a different sensory modality such as visual and auditory sensations. [11] gives examples of modalities such as visual signals like videos, vocal signals like sounds, and natural language like text, with the latter being the main focus of this work. Research in this field deals with datasets such as videos with both visual and auditory signals.

In [11], Baltruaitis et al. lists the five core challenges of multimodal machine learning as Representation, Translation, Alignment, Fusion, and Co-Learning. This work focuses on the Fusion challenge. There are two main Feature fusion schemes, identified by [12], known as early fusion and late fusion. Early fusion is described as occurring in the feature space before any learning is performed. In early fusion, feature sets are combined before being given to a single classifier. Late fusion is the process of training separate learners on each unimodal representation and then combining these learned scores later to a multimodal representation which can then be used to produce a classification. The work in this thesis focuses on early fusion of modalities. As noted in [11] the benefits of multimodal machine learning can be seen in the increase in performance due to the exploitation of complementary information found in different modalities such as the speech information found in visual images of someone speaking along with an audio signal. Along with the complementary information, there is the benefit of the information provided by the other modalities if one modality is degraded. These two points show that being able to combine modalities will create much more robust systems than their unimodal counterparts. The process of feature fusion used in this work is described in more depth in Chapters 3 and 4.

1.5 Datasets

The dataset used in this thesis, the CASIS-25 dataset, is a subset of the CASIS-1000 dataset [2, 19]. The CASIS-1000 dataset is composed of online blog entries from 1000 authors. For each author in this dataset, there are 4 writing samples for a total of 4000 writing samples in the entire dataset. The CASIS-25 dataset consists of the first 25 authors of the CASIS-1000 dataset. Given the 25 authors with 4 writing samples per author, the CASIS-25 dataset consisted of a total of 100 writing samples. The average number of sentences for a writing sample is 12, and the maximum and minimum number of sentences are 37 and 4. On average each writing sample has 390 words with a maximum number of words in a given sample being 963 and a minimum of 114. Finally, there is an average of 25 words per sentence with a minimum of 2 words in a sentence and a maximum of 116 in a sentence. The difficulty of this dataset comes from the relatively small number of samples for each author and the relatively small size of the samples. This dataset is used to obtain the results in sections 2, 3, and 4.

1.6 Scope of Work

This work in this thesis seeks to examine the performance of the Sentiment Analysis, LIWC, and Topic Model representations for Authorship Attribution. Each representation will be evaluated individually as well as in combination with one or more of the other representations. Each of the representations will first be created for each writing sample by extracting the different features sets from the CASIS-25 dataset. Each individual feature set will be evaluated in a small-scale Authorship Attribution system. Subsets of these features will be evaluated using the feature selection method known as GEFeS. Then the combinations of the feature sets will be evaluated using the combination of GEFeS and the feature fusion techniques derived from Multimodal Machine Learning concepts. The remainder of the work is as follows. Chapter 2 will discuss the performance of the Sentiment Analysis feature set and its performance. Chapter 3 will discuss the LIWC feature set and its performance alone and in combination with the Sentiment Analysis features. Chapter 4 will discuss the Topic Model feature set and its performance alone and in combination with both the Sentiment Analysis and LIWC feature sets. Finally, Chapter 5 will provide a summary of the work accomplished as well as future directions of the work.

Chapter 2

Sentiment Analysis

OpinionFinder 2.0 [13, 14, 15, 16] was used to create a representation based on Sentiment Analysis. This program goes through a few steps to determine the sentiment of a given document. This program first uses the Stanford POS Tagger to tokenize the text and then uses a set of dictionaries to label the tokens found. After this it uses a set of classifiers to create more robust sentiment labels. The features [17] used in this thesis only rely on the output from the tagger and the labels generated from the internal dictionaries and were generated using feature engineering techniques [18]. The labels used to generate the features come from the ”subjclueslen1polar.tff” [14, 15, 16] dictionary. This dictionary provides two sets of labels which tag words in terms of their subjectivity and their polarity. Table 2.1 shows each of the subjective and polar labels provided in the dictionary. Each token matched in the dictionary is labeled with one of the two subjectivity labels and one of the six polarity labels.

”subjclueslen1polar.tff” Labels						
Subjectivity	strongsubj	weaksubj				
Polarity	strongpos	weakpos	neutral	weakneg	strongneg	both

Table 2.1: The Subjectivity and Polarity Labels Provided by ”subjclueslen1polar.tff”.

In Table 2.1 the first row shows the two Subjectivity labels which are ”strongsubj” and ”weaksubj”. The Subjectivity labels denote whether the found token is either strongly subjective or weakly subjective respectively. The second row then shows each of the Polarity labels which are ”strongpos”, ”weakpos”, ”neutral”, ”weakneg”, ”strongneg”, and ”both”. The Polarity labels denote whether the found token is positive, neutral, negative or both in terms of polarity. Using the OpinionFinder labels a total of 176 features are created. The first eight features are the individual probabilities of each label occurring. Specifically, each of the first eight features are P(strongsubj), P(weaksubj), P(strongpos), P(weakpos), P(neutral), P(weakneg), P(strongneg), and P(both). The next 24 labels consisted of each combination of conditional probability in the form of P(Subjectivity — Polarity) and P(Polarity — Subjectivity). An example of each respectively would be P(strongsubj — weakpos) and P(strongneg — weaksubj). With two Subjectivity labels and six Polarity labels there are twelve features of the form P(Subjectivity — Polarity) and twelve more features of the form P(Polarity — Subjectivity) for a total of 24 conditional probability features. The remaining 144 features are counted features based on the number of state transitions that occur in a given document. In this context a state is the combination of Subjectivity and Polarity label of the form (Subjectivity, Polarity). An example of a single feature would be the number of transitions from (strongsubj, strongpos) to (strongsubj, weakneg).

2.1 Genetic and Evolutionary Feature Search

As described in the Introduction, GEFeS is a feature selection algorithm that uses a Steady-State Genetic Algorithm to evolve feature masks that turn features on or off. GEFeS starts with a population of 100 feature masks. Each of these feature masks are mutants generated from a feature masks of all one's which uses all of the features. An initial mutant rate is used to remove a certain percentage of features from this feature mask of all one's to produce the mutants. The initial population is evaluated using stratified 4-fold cross validation which is described further in the Experiment Section. After the initial population is evaluated the evolutionary algorithm loop begins. For each generation two features masks are chosen using tournament selection with a tournament size of 2 to be the parents. First two feature masks are chosen at random and the feature mask with the highest fitness score is chosen to be the first parent. The fitness function used to determine the fitness of the feature masks is described in more detail in the next paragraph. This process is repeated again to choose the second parent. With these two parent feature masks a single offspring feature mask is generated using uniform crossover and then mutated with a mutation rate of 2%. Then this offspring feature mask will replace the feature mask with the worst fitness from the population. This is repeated for 14900 times to create 14900 offspring feature masks. There is a total of 15000 function evaluations between the initial population of 100 and the 14900 offspring feature masks created.

The fitness function is defined as $Fitness(fm_i) = \alpha_i - w\beta_i$. fm_i is a feature mask, α_i is the accuracy of fm_i, and β_i is the percentage of the features used by fm_i. The last variable w is the feature reduction weight where $0 \geq w \geq 1$. This value tells how strongly the fitness function should value the reduction of features compared to the accuracy. The use of this fitness function has two benefits. With an emphasis on removing features while also retaining a relatively high accuracy GEFeS will remove unnecessary or noisy features which do not provide much information while allowing only the most salient features to propagate up. This fitness function also gives the SSGA direction in the search space. As shown in the results, a purely accuracy-based fitness function, or a feature reduction weight of 0.0, will tend to use roughly the same amount of features as the initial mutant rate and will not perform as well as other feature reduction weights.

2.2 Experiment

To examine the performance of the Sentiment Analysis features for Authorship Attribution, features are extracted from a set of documents and then evaluated on three classifiers using stratified four-fold cross validation. The high-level classification pipeline is shown in Figure 2.1. The set of documents are passed through the feature extraction module to retrieve the Sentiment Analysis features. Then these features are preprocessed and passed to the classifier to determine the most probable author.

The set of documents comes from the CASIS-25 dataset which is described above in the introduction. The preprocessing consisted of converting the extracted feature vectors into TF-IDF representations, standardization, and then normalization. The three classifiers used in this thesis are the MLP (multilayer perceptron), and two variants of a support vector machine. The MLP is composed of a single hidden layer with 100 units and rectified linear unit activation functions. The two variants of the support vector machine are referred to as the LSVM (linear support vector machine) and the RBFSVM (radial basis function support vector machine). The

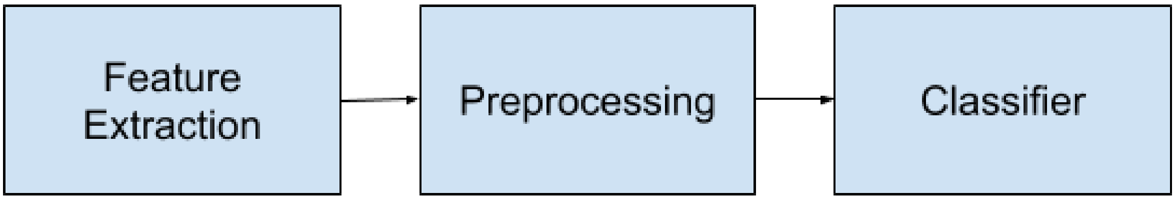

Figure 2.1: Classification Pipeline

LSVM employs a linear kernel while the RBFSVM employs a radial basis function kernel. The use of stratified four-fold cross validation ensures that each classifier is trained and evaluates each writing sample. Since each author has four writing samples, each test fold will contain one writing sample from each author while the training fold will contain the remaining three writing samples for each author.

2.3 Results

The results of the baseline evaluation for each classifier can be seen in Table 2.2. The first row shows the classifier and the second row shows the cross-validation accuracy. The LSVM performs the worst at 18% accuracy and the RBFSVM performs only slightly better at 19% accuracy. Since the MLP classifier is non-deterministic and can produce different weights after each training run, its performance was examined over 30 instances. The maximum accuracy achieved from the MLP is 22% and the average accuracy over the 30 instances is 19% shown in parentheses.

Classifier	Accuracy
LSVM	18%
RBFSVM	19%
MLP	**22% (19%)**

Table 2.2: The Baseline Accuracy of Sentiment Analysis Features on the CASIS-25 Dataset.

Using GEFeS, feature masks are generated for each classifier on the CASIS-25 dataset. The initial population consisted of 100 feature masks which were created using an initial mutant rate of 50% so each feature mask used roughly 50% of the total 176 features. Tables 2.3, 2.4, and 2.5 show the performance of the feature masks generated with GEFeS for the MLP, RBFSVM, and LSVM. The first column in each of these tables shows the GEFeS feature reduction weights that were tested. For each a weight a total of 30 masks were generated. The second column shows the maximum and average accuracy that was achieved for a given weight. The average accuracy is displayed in parentheses. The third column shows the percentage of the total features that were used. The value in parentheses is the average percent of features used by masks generated from that feature reduction weight. The other value is the percent of features used by the mask that achieved the maximum accuracy.

Feature Reduction Weight	Accuracy	Features Used
0.0	49.0%(46.6%)	39.8%(43.9%)
0.1	51.0%(47.8%)	36.9%(37.1%)
0.3	52.0%(49.6%)	29.0%(29.0%)
0.5	**53.0%(49.9%)**	**24.4%(24.5%)**
0.7	52.0%(48.7%)	22.2%(21.9%)
0.9	52.0%(48.3%)	17.6%(17.9%)
1.0	51.0%(48.1%)	16.5%(17.5%)

Table 2.3: The Comparison of Different Feature Reduction Weights in the GEFeS-MLP Classifier.

The results for the GEFeS-MLP classifier are shown in Table 2.3. Using feature reduction weights ranging from 0.0 to 1.0, GEFeS generates features masks which use anywhere from 16.5% to 43.9% of the total 176 features. The accuracy achieved by the feature masks on average are greater than that of the baseline MLP accuracy. A feature reduction weight of 0.5 produced the best performing feature mask. This mask achieved a maximum accuracy of 53% by using 24.4% of the total features. This is an increase of 31 percentage points from the MLP baseline accuracy. The results for the GEFeS-RBFSVM classifier are shown in Table 2.4. The feature mask produced with a feature reduction weight of 0.5 was able to achieve an accuracy of 54% while only using 21% of the total features. Once again, this is more than twice as accuracy as the baseline RBFSVM which used all Sentiment Analysis features. Finally, the GEFeS-LSVM results are shown in Table 2.5. Feature masks produced for this classifier achieve an accuracy ranging from 50.6% all the way to the highest accuracy achieved using Sentiment Analysis features which was 57%. The best performing feature weight here was the weight of 0.7. A mask produced by this weight achieved the highest accuracy by only using 21.6% of the total features.

Feature Reduction Weight	Accuracy	Features Used
0.0	51.0%(49.5%)	39.8%(42.7%)
0.1	53.0%(51.1%)	34.7%(36.2%)
0.3	54.0%(50.9%)	32.4%(29.1%)
0.5	**54.0%(50.9%)**	**21.0%(23.4%)**
0.7	53.0%(50.2%)	20.5%(21.1%)
0.9	51.0%(48.5%)	18.8%(17.5%)
1.0	52.0%(48.7%)	18.2%(17.0%)

Table 2.4: The Comparison of Different Feature Reduction Weights in the GEFeS-RBFSVM Classifier.

Feature Reduction Weight	Accuracy	Features Used
0.0	54.0%(51.1%)	44.3%(42.4%)
0.1	56.0%(52.3%)	30.1%(35.5%)
0.3	56.0%(53.0%)	24.4%(28.2%)
0.5	56.0%(53.2%)	21.0%(24.5%)
0.7	**57.0%(52.8%)**	**21.6%(20.9%)**
0.9	54.0%(51.0%)	15.9%(16.8%)
1.0	54.0%(50.6%)	17.6%(16.2%)

Table 2.5: The Comparison of Different Feature Reduction Weights in the GEFeS-LSVM classifier.

Finally, the most consistent features were found. This involved examining the best performing feature mask from each of the 30 runs, a total of 30 feature masks, and determining how often each feature occurs out of the 30 masks. The features used in the masks generated from GEFeS-LSVM with a feature reduction weight of 0.7 since it produced the highest accuracy using Sentiment Analysis features. 47 features did not occur at all in the best performing feature masks while the other features ranged anywhere from 3% to 100%. Table 2.6 shows the 6 features that occurred 100% of the time out of all of the 30 runs. Each of these state transition features occurred in the 30 of the best performing feature masks and are considered to be the most stable. A full list of the features and their consistency is shown in Appendix A.

Feature Name
(StrongSubj,StrongPos) → (StrongSubj,StrongPos)
(StrongSubj,Neutral) → (StrongSubj,StrongPos)
(StrongSubj,Neutral) → (WeakSubj,StrongPos)
(StrongSubj,Neutral) → (WeakSubj,Neutral)
(WeakSubj,Neutral) → (WeakSubj,WeakPos)
(WeakSubj,WeakNeg) → (WeakSubj,WeakNeg)

Table 2.6: Most Consistent Sentiment Analysis Features.

Chapter 3

Linguistic Inquiry and Word Count

The LIWC program [7] is a sophisticated application that is able to extract information from text to provide insight about its author from a psychological standpoint. The heart of the LIWC program is the internal dictionary. This dictionary is comprised of almost 6400 words that are placed in different hierarchical categories. Similar to OpinionFinder, LIWC processes each word in a document individually to find matches of words in the document to words in the internal dictionary.

The first four features extracted by LIWC are the word count of the document as well as three language metrics which consist of the average number of words per sentence, the number of words with greater than six letters, and the number of words found that are actually matched in the dictionary. The next 85 features come from the different categories in the LIWC dictionary. Each LIWC category is a set of hand-picked words. These categories can range from objective categories like punctuation or articles to more subjective categories like positive or negative emotion. The features extracted using the different categories are based on the percentages of words found in the given category out of the total number of words found in the document. These categories are hierarchical as well. The category Total pronouns is divided into Personal pronouns and Impersonal pronouns, and the Personal pronouns category is divided further into subcategories. If a word is found in a subcategory of Personal pronouns then the percentages for that given subcategory, the Personal pronouns category, and the Total pronouns category are all increased.

The last four features produced by LIWC are summary variables which come from proprietary algorithms that work on other LIWC variables. These four summary variables are

Analytical thinking, Clout, Authenticity, and Emotional tone each of which are on a range of 0 to 100. Analytical thinking [20] is derived from eight function word categories. A higher use of auxiliary verbs, pronouns, adverbs, conjunctions, and negations suggests the writer generally thinks in a narrative style and about personal experiences. This is considered to be on the lower end of the spectrum of Analytical thinking. A higher use of articles and prepositions, however, indicated more categorical thinking or more complexly organized concepts. This is considered to be on the higher end of the spectrum of Analytical thinking. Clout [21] denotes the relative rank or status in a group based on pronoun usage. Higher ranked individuals or individuals with higher Clout scores tend to be more focused on others which is shown by their higher usage of first-person plural and second-person singular pronouns. While, lower ranked individuals tend to be more focused on self which is shown in their usage of more first-person singular pronouns. Authenticity [22] shows whether the author is honest, high in Authenticity, or deceptive, low in Authenticity. This is derived from a number of other LIWC categories but according to [23] deceptive authors are characterized by lower cognitive complexity, used fewer self-references and other-references, and had a larger number of negative emotion words in their writings. Finally, Emotional Tone [24] denotes whether the document has a positive or negative overall tone. This is scored is produced from the two of the categorical features known as Positive Tone and Negative Tone. When this variable is high the overall tone of the document is positive while when this variable is low the overall tone of the document is negative.

In total LIWC provides 93 features from a given document. While these features are for the most part just word frequencies, LIWC claims that the hand-picked word categories can point to a deeper psychological insight of the author. This is especially true in the case of the summary variables which are built from other LIWC features.

3.1 Experiment

The evaluation process was similar to that of the process describe in Section 2 and the same classification pipeline shown in Figure 2.1 is used as well. The dataset used was the CASIS-25 dataset. LIWC features are extracted from these documents and evaluated on the MLP, RBFSVM, and LSVM classifiers using stratified 4-fold cross validation. The same preprocessing method is used wherein the feature vectors extracted from the CASIS-25 dataset are transformed in TF-IDF representation, standardized, and then normalized. GEFeS is also used as a feature selection method to look at different subsets of the total feature set. However, in this chapter, the performance of the fusion of the LIWC feature set and the Sentiment Analysis feature set is examined along with the performance of the LIWC feature set alone. This feature fusion process is shown in Figure 3.1. First both the LIWC feature vector and the Sentiment Analysis Feature vector are extracted from a document. Then the two feature vectors are concatenated as a single feature vector before being preprocessed and then classified.

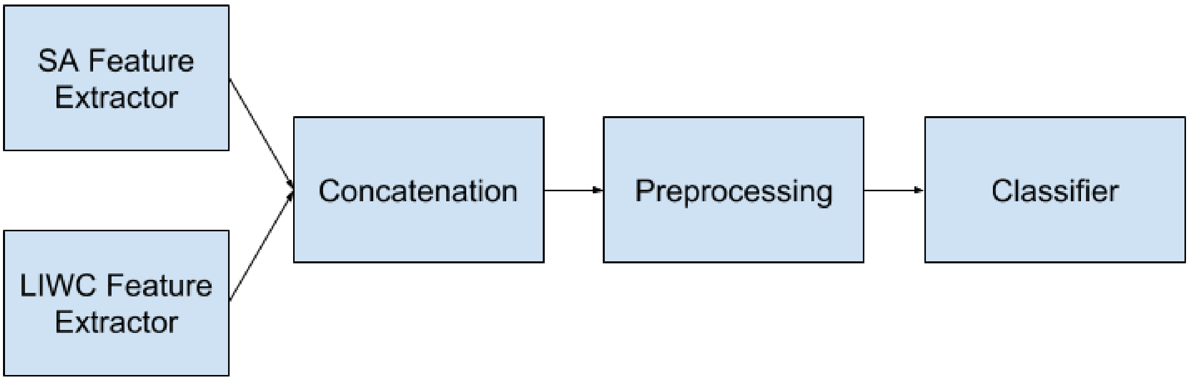

Figure 3.1: Feature Combination Process

3.2 Results

The results of the stratified 4-fold cross validation accuracy for each classifier are shown in Table 3.1. Each classifier uses all 93 LIWC features. The RBFSVM performs the worst out of the three with an accuracy of 71%. This however is a large improvement compared to the Sentiment Analysis features. Next the results of the MLP are examined for 30 instances of the MLP classifier due to its non-deterministic behavior. Out of the 30 the average accuracy is 73%, shown in parentheses, and the maximum accuracy achieved is 78% on a single instance. Finally, the LSVM performs the best out of the three classifiers with an accuracy of 84%.

The accuracy of the combination of the feature sets is shown in Table 3.2. The MLP classifier performed the worst on average at 57% accuracy, but it did achieve a maximum accuracy

Classifier	Accuracy
RBFSVM	71%
MLP	78% (73%)
LSVM	**84%**

Table 3.1: The Baseline Accuracy of LIWC Features on the CASIS-25 Dataset.

of 62% on a single run. The RBFSVM achieved an accuracy in between the MLP classifier's maximum and average accuracy at 58%. The LSVM performed this best with an accuracy of 68%. The combination of the two features sets did perform better than the Sentiment Analysis features alone as shown in Table 2.2. It appears that the LIWC features were able to provide some extra information that the Sentiment Analysis features did not capture. However, the combination performed worse than the LIWC features alone shown above in Table 3.1. The Sentiment Analysis features added too much noise compared to the LIWC features.

Classifier	Accuracy
RBFSVM	58%
MLP	62% (57%)
LSVM	**68%**

Table 3.2: The Baseline Accuracy of the Concatenation of Sentiment Analysis and LIWC Features on the CASIS-25 Dataset.

The results for GEFeS-LSVM features masks on the LIWC feature set is shown in 3.3. An initial mutant rate of 50% was used and similar to the other GEFeS tables the feature reduction weight is shown in the first column, in the accuracy column the maximum and average accuracy are shown, and in the Features Used column the percentage of features used for the best performing feature mask and the average percentage of features used is shown. This table also lists the equivalence classes for Accuracy and Percentage of features used. With 30 instances for each feature reduction weight, equivalence classes can be derived by determining which feature reduction weights produce feature masks that have a statistically significant difference in terms of accuracy or Percentage of features used. Feature reduction weights of 0.0, 0.1, and 0.3 are all in Equivalence Class I in terms of accuracy. This means that they all produce feature masks that provide the highest accuracy but have no statistically significant different between them. A feature reduction weight of 1.0 is in Equivalence Class I in terms of percentage of features used because it was able to use the least number of features. However, it is in Equivalence Class V in terms of accuracy which means that while it was able to reduce the amount of features used the accuracy suffered.

Feature Reduction Weight	Equivalence Class		Accuracy	Features Used
	Accuracy	Features		
0.0	I	VII	96.0% (95.4%)	52.6% (59.7%)
0.1	I	VI	96.0% (95.4%)	47.3% (57.5%)
0.3	**I**	**V**	**97.0% (95.1%)**	**41.9% (47.8%)**
0.5	II	IV	96.0% (93.7%)	33.3% (37.2%)
0.7	III	III	94.0% (92.6%)	25.8% (29.9%)
0.9	IV	II	94.0% (92.1%)	23.6% (28.4%)
1.0	V	I	93.0% (90.9%)	21.5% (25.8%)

Table 3.3: The Comparison of Different Feature Reduction Weights in the GEFeS-LSVM Classifier on LIWC Features.

The feature set combination method can also be applied when using GEFeS. Table 3.4 shows the results of using the combined feature sets with GEFeS-LSVM. Similar to the previous GEFeS-LSVM above an initial mutant rate of 50% is used. In this table feature reduction weights of 0.3-0.9 are all in Equivalence Class I in terms of accuracy. Using more features with the combined feature set required a greater feature reduction weight than when only the LIWC feature set is used. The combined feature set was also able to increase the accuracy slightly as well. Feature reduction weights 0.3-0.9 were able to increase the accuracy from 97% to 98%.

Feature Reduction Weight	Equivalence Class		Accuracy	Features Used
	Accuracy	Features		
0.0	III	VII	96.0% (94.0%)	42.8% (47.6%)
0.1	II	VI	97.0% (95.3%)	37.2% (41.2%)
0.3	I	V	98.0% (96.1%)	34.2% (34.2%)
0.5	I	IV	98.0% (96.2%)	29.0% (29.3%)
0.7	I	III	98.0% (96.5%)	24.5% (26.6%)
0.9	**I**	**II**	**98.0% (95.9%)**	**21.9% (23.4%)**
1.0	II	I	98.0% (95.6%)	23.8% (22.6%)

Table 3.4: The Comparison of Different Feature Reduction Weights in the GEFeS-LSVM Classifier on the Concatenation of Sentiment Analysis and LIWC Features.

Once again, the most consistent features were examined for the LIWC features. The same procedure used to determine the most consistent features for Sentiment Analysis was used to determine the most consistent LIWC features. This was only performed for the LIWC features alone in this chapter and not for the combination of LIWC and Sentiment Analysis features. In this case GEFeS-LSVM with a feature reduction weight of 0.3 was examined since it performed the best on LIWC features alone. 28 of the 93 LIWC features were unused in all of the best performing features masks from the 30 runs. The remaining features are used anywhere from 3% to 100%. Table 3.5 shows the 12 features that were used 100% of the time in each of the 30 best feature masks. A full list of features and their consistency is shown in Appendix B.

Feature Name
Word Count
Words per Sentence
Six Letter
Female Referents
Insight
Cause
Power
Work
Money
Period
Dash
Apostrophe

Table 3.5: Most Consistent LIWC Features.

Chapter 4

Topic Modeling

Topic Models are generative models based on the idea that documents are created from a mixture of topics. The procedure for generating documents is described in [25] as choosing a distribution over topics, which are distributions over words. In order to generate new words for a document, a topic is sampled from the distribution of topics and then a word is sampled from the chosen topic. This is the process for generating new documents, however, given a set of already generated documents this process can be reversed using statistical techniques to infer topics from the set of given documents.

MALLET [26] is a program that implements this reverse procedure. MALLET is able to infer topics from a given set of documents by utilizing an algorithm described in [25] which uses a technique known as Gibbs sampling. This program takes in a set of documents as input and will return the topic that each word found in the set of documents is most likely associated with as well as the distribution of topics within each document. It also outputs the top words in each topic to give an idea of what the topic is about.

In order to use Topic Models for Authorship Attribution, the MALLET program is used to extract a given number of topics in an unsupervised fashion from a group of documents. After extracting the topics from a group of documents, each of the documents can be characterized as a feature vector that consists of the frequency of each topic in the document. This method is similar to that of the other feature extractors described in this thesis in that words are grouped into higher level categories and the frequencies of these categories are used as features. The difference in the case of Topic Modeling is that categories are discovered in an unsupervised way. The categories in the Sentiment Analysis and LIWC feature extractors are explicitly defined.

The benefit of the implicit categories derived from Topic Models is that each topic or category will be specific to the given set of documents. Each word found in the set documents will likely be present in at least one category and can provide information to better attribute the document. In the case of hand-crafted categories with explicitly defined words, such as the Sentiment Analysis and LIWC features, there could be many words that do not match to any category in the dictionary. This means that fewer words will be available to provide information to attribute the document. On the other hand, the drawback is that common problem of Topic Models that is actually choosing the correct number of topics for a given set of documents. Choosing too few topics will produce topics that are general and are not able to characterize the documents as well while choosing too many topics will create topics that capture the same information and will be less meaningful individually.

4.1 Multimodal Preprocessing

As discussed in the Introduction, most research in the area of multimodal machine learning focuses on combining different forms of media such as text, audio, and visual data. However, these concepts of traditional multimodal machine learning can be applied to the area of multimodal authorship attribution were each mode can be viewed as a set of features or representation. Handling multiple modalities when combining different mediums, such as video, audio and text, is particularly useful because each medium is going to have its own representation and most likely will have distinct statistical properties. This thesis only focuses on text which only has the one representation of counts and frequencies. However based on the nature of each feature set they could have their own distinct statistical properties even though they share the same representation. Each feature set in this thesis characterizes the text in a different way whether it is looking at the sentiment of the text with the sentiment analysis features or the psychological state of the author with the LIWC features. This is achieved by counting the words found in categories. These categories can be explicitly defined, as in the sentiment analysis and LIWC features, with dictionaries of words or implicitly defined, as in the Topic Modeling features, that create categories in an unsupervised fashion. Given the number of words and chosen of words in each category, each characterization could potentially have its own distinct underlying statistical distribution. While not fully explored in this thesis there appears to be some evidence of this as seen in later in the results. A similar classification pipeline and fusion method are used as described in previous chapters in Figures 2.1 and 3.1. This pipeline consists of extracting the different feature vectors, preprocessing the feature vectors, and then passing them through the classifier. The fusion method is a simple concatenation of feature vectors. Previously feature vectors were fused after the extraction and before the preprocessing so the preprocessing module treated the combined feature vector as if it were unimodal. In this chapter the fusion is pushed further into the pipeline and the feature vectors are fused after the preprocessing and before the classification. In this approach, illustrated in Figure 4.1 each representation is preprocessed separately before being combined. This seems like a small change however this method has improved the performance of all combination of feature sets used in this thesis as seen in the results. Each of these methods would be classified as early fusion based on the descriptions in the Introduction since the fusion occurs in the feature space. However, there is clearly some benefit to recognizing the different modalities in each feature set and handling that in the preprocessing step.

4.2 Experiment

Once again, the evaluation method used in this section is similar to that of the previous sections. The CASIS-25 dataset is still used as the dataset. The preprocessing process is the same in that the features are transformed into tf-idf representation and then standardized and normalized. Then stratified 4-fold cross validation is used to produce an accuracy score on the MLP, LSVM, and RBFSVM classifiers. The main difference is the modified feature fusion process discussed in the previous section and shown in Figure 4.1.

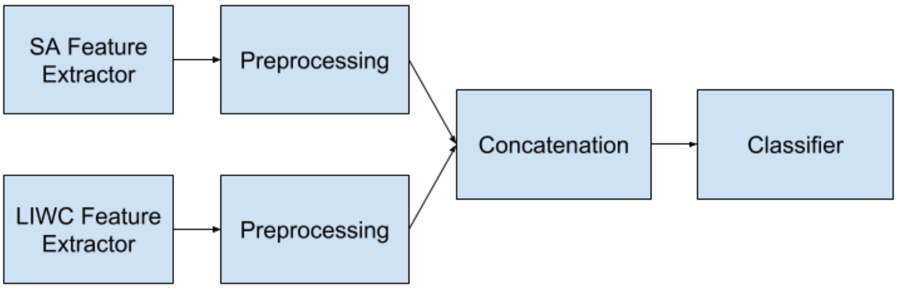

Figure 4.1: Modified Feature Combination Process

4.3 Results

When extracting Topic Model features, the number of topics is a key factor in getting topics with the most information. Choosing too low of a number will produce a few general topics while choosing too many topics will result in a large number of topics with a high potential for overlap. To find the best number of topics to extract and use as features for the CASIS-25 dataset, a number of topics was chosen and then evaluated using the evaluation process described above. This accuracy was then compared to the accuracy of other topic numbers to determine which produced the highest accuracy. Table 4.1 shows the results of this process. The first column shows the number of topics used and the other three columns show the evaluation accuracy from the three base classifiers before any feature selection is performed. The MLP accuracy once again was evaluated over 30 instances due to the non-deterministic nature of the MLP. The maximum accuracy achieved out of the 30 instances is shown as well as the average accuracy which is shown in parentheses. Starting with 10 topics, the number of topics that was evaluated increased in increments of 5 up until 50 topics. After 50 topics was evaluated, a decrease in accuracy was noticed so the increment was increased to 10 just to ensure the trend downward would continue. 45 topics produced the best accuracy for each of the classifiers at 89% accuracy.

Number of Topics	MLP	RBFSVM	LSVM
10	49% (46.0%)	40%	41%
15	65% (62.8%)	62%	61%
20	79% (77.3%)	76%	76%
25	85% (82.9%)	78%	79%
30	86% (84.8%)	84%	84%
35	82% (80.4%)	82%	82%
40	89% (87.9%)	86%	88%
45	**89% (87.4%)**	**89%**	**89%**
50	78% (77.1%)	80%	77%
60	76% (75.0%)	76%	75%
70	67% (66.2%)	70%	66%
80	60% (58.7%)	63%	63%
90	59% (56.4%)	63%	61%
100	52% (48.6%)	48%	49%

Table 4.1: Evaluation of Different Numbers of Topics on the Three Base Classifiers.

Next the use of GEFeS on the new feature set was investigated. In this section the RBFSVM classifier produced the best results for the Topic Modeling features so it is the only one shown in Table 4.2. This Table has similar columns to that of the GEFeS table in the previous section. The feature reduction weight as well as the equivalence class, accuracy, and percent of features used are shown. Feature reduction weights of 0.0 and 0.1 are in Equivalence Class I in terms of accuracy and in Equivalence Classes VI and V respectively in terms of feature usage. By removing roughly 33.3% of the features the best performing masks generated by these feature reduction weights were able to achieve an accuracy of 95%.

Next before the use of GEFeS on the combined feature sets is shown, the difference between the two preprocessing methods is shown in Table 4.3. This table specifically shows the improvement of accuracy achieved on the different combinations of each feature set shown in this thesis using the combination method described in Figure 3.1 and the new combination method described in Figure 4.1. The first column shows the feature sets that are being combined and the second column shows the total amount of features being used. Next the 3 column shows the classifier and last column shows the accuracy achieved. The unimodal column is

Feature Reduction Weight	Equivalence Class		Accuracy	Features Used
	Accuracy	Features		
0.0	I	VI	95.0% (95.0%)	66.7% (70.0%)
0.1	**I**	**V**	**95.0% (95.0%)**	**66.7% (66.5%)**
0.3	II	IV	95.0% (94.5%)	66.7% (64.4%)
0.5	III	III	92.0% (91.0%)	57.8% (55.6%)
0.7	III	III	92.0% (92.1%)	57.8% (55.8%)
0.9	IV	II	89.0% (87.3%)	53.5% (51.6%)
1.0	IV	I	89.0% (87.1%)	53.3% (51.2%)

Table 4.2: The Comparison of Different Feature Reduction Weights in the GEFeS-RBFSVM Classifier on Topic Model Features.

the accuracy achieved. using the process shown in Figure 3.1 and the multimodal column is the accuracy achieved using the process shown in Figure 4.1. Except for the SA and LIWC feature combination using MLP, the multimodal preprocessing technique was able to achieve a higher accuracy than the unimodal preprocessing technique. Using an RBFSVM classifier and the multimodal processing on the LIWC and TM features produces 93% accuracy which is 15% greater than that of its unimodal counterpart and only 3% less than the accuracy of the GEFeS-RBFSVM on the TM feature set as shown in Table 4.2. Then adding the SA features the RBFSVM classifier and the multimodal processing technique increases the accuracy by one percentage point to 94% which is 21% greater than the accuracy of the unimodal preprocessing and only 1% less than that of the GEFeS-RBFSVM on the TM features.

Feature Set	Total Features	Classifier	Accuracy	
			Unimodal	Multimodal
SA and LIWC	269	MLP	63%	60%
		RBFSVM	58%	62%
		LSVM	68%	**74%**
SA and TM	221	MLP	67%	86%
		RBFSVM	72%	**90%**
		LSVM	80%	88%
LIWC and TM	138	MLP	87%	92%
		RBFSVM	88%	**93%**
		LSVM	86%	91%
SA, LIWC and TM	314	MLP	78%	93%
		RBFSVM	83%	**94%**
		LSVM	82%	**94%**

Table 4.3: The comparison of preprocessing methods on each combination of feature sets.

Finally, the results of GEFeS on the combination of each feature set presented in this thesis are shown in Table 4.4. As described above the RBFSVM performed the best out of the classifiers after applying GEFeS so it was the only table featured. The combination of the GEFeS feature masking and multimodal preprocessing was able correctly classify every instance in the CASIS-25 dataset. Feature reduction weights of 0.1, 0.3, and 0.5 are all in Equivalence Class I in terms of accuracy even though every feature reduction weight eventually produced a feature mask that achieved 100% accuracy. The feature masks ended up using anywhere from 47.5% to 17.8% of the total features.

Feature Reduction Weight	Equivalence Class		Accuracy	Features Used
	Accuracy	Features		
0.0	II	VI	100.0% (99.4%)	47.5% (50.4%)
0.1	I	V	100.0% (99.7%)	27.4% (28.9%)
0.3	I	IV	100.0% (99.7%)	22.6% (25.4%)
0.5	**I**	**III**	**100.0% (99.4%)**	**22.9% (23.2%)**
0.7	II	II	100.0% (98.9%)	19.4% (20.1%)
0.9	II	I	100.0% (98.8%)	18.8% (19.2%)
1.0	II	I	100.0% (98.7%)	17.8% (18.9%)

Table 4.4: The comparison of different feature reduction weights in the GEFeS-RBFSVM classifier on the combination of Sentiment Analysis, LIWC, and Topic Model features using multimodal preprocessing techniques.

Feature consistency for the Topic Model features looked much different than that of the Sentiment Analysis and LIWC features. Feature masks generated by GEFeS-RBFSVM with a feature reduction weight of 0.1 were examined to determine the feature consistency. Unlike the Sentiment Analysis and LIWC features where features for the most part varied between feature masks, except for a select few, 28 of the Topic Model features occurred 100% of the time in the 30 masks. Aside from the 28 features that occurred consistently there were two other features used where one of the features occurred 75% of the time and the other 25% of the time. The remaining 15 features were not used at all. Unlike the Sentiment Analysis and LIWC features the Topic Model features seemed to converge fairly consistently to the same masks each time. A full list of features and their consistency is shown in Appendix C.

Chapter 5

Conclusions & Future Directions

In this thesis three different representations for textual data were examined for the purposes of Authorship Attribution. The first was a representation derived from a Sentiment Analysis feature set, the second was derived from the LIWC feature set, and the third was derived from a Topic Model feature set. Features from these sets were extracted from the CASIS-25 dataset to test their performance in a small-scale Authorship Attribution system. Each feature set was tested on its own and in combination with the other features. A feature selection method known as GEFeS, which is based on a steady-state genetic algorithm, was used to find the most salient features in each set, and feature fusion techniques derived from Multimodal Machine Learning were used to combine the different feature sets.

The results show that Sentiment Analysis features alone perform poorly even with the help of feature selection. This aligns with the results in [5], however, [6] shows that there could still potentially be some usefulness in using Sentiment Analysis for Authorship Attribution. LIWC features alone however prove to be quite informative for Authorship Attribution and with the help of feature selection can provide close to 100% accuracy. The first feature fusion attempts with LIWC and Sentiment Analysis performed poorly, but with feature selection the combined feature sets were able to outperform LIWC features alone. Topic Model features performed the best on their own before feature selection or fusion. A modification to the feature fusion process improved the technique overall, and the combination of the feature selection and feature fusion using all three feature sets was able to correctly classify each of the authors in the give dataset.

In terms of scaling a single feature mask generated on a smaller dataset does not provide the same performance as shown in Table 5.1. The best performing feature mask generated for

Sentiment Analysis, LIWC, and Topic Model features is able to achieve 100% cross validation accuracy. However when this feature mask is applied to larger datasets such as the CASIS-50 dataset that contains 50 authors it only achieves 59.5% accuracy and when applied to the CASIS-100 dataset that contains 100 authors it only achieves 50.5% accuracy. The feature mask generation process itself turns out to be much more viable on larger datasets as seen in Table 5.2. When applied to the CASIS-25 dataset the process can classify authors correctly 100% of the time. When applied to the CASIS-50 dataset GEFeS-RBFSVM with a weight of 0.5 is able to achieve 93.5% accuracy and when applied to CASIS-100 dataset GEFeS-RBFSVM with a weight of 0.1 is able to achieve 79.2% accuracy. The comparison of the results shows that the process is much more scalable.

Dataset	CASIS-25	CASIS-50	CASIS-100
Accuracy	100%	59.5%	50.5%

Table 5.1: The scaling of a feature mask.

Dataset	CASIS-25	CASIS-50	CASIS-100
Accuracy	100%	93.5%	79.2%

Table 5.2: The Scaling of the Process.

Future work could involve improving the feature mask generation process and applying it to larger subsets of the CASIS-1000 dataset. In terms of feature fusion, only early fusion was examined in this work. Late fusion, which would involve merging the results of different classifiers trained individually on each feature set, could be examined to see if this would provide an improvement over early feature fusion which only occurs in the feature space. Other improvements could be provided by looking into feature weighting instead of feature masking. The current binary feature masks generated by GEFeS turn features on or off. Feature weighting techniques [29] could be used generate weights for each feature so that instead of turning a feature completely off certain features would just provide less weight in terms of the information used for classification. Other future work will involve the development and testing of DiNEH (Distributed Neuro-Evolutionary Hybrid). Similar to GEFeS, DiNEH will generate high performance feature masks for Authorship Attribution. The difference is that GEFeS deals with a single classifier trying to classify N authors. DiNEH is a more scalable approach that splits this problem into N classifiers where each classifier is trained to identify a single author as a two-class problem. Adding a new author to a GEFeS system would mean starting over and retraining the single classifier and producing another completely different set of feature masks. Adding a new author to DiNEH also involves training a new classifier but all of the classifiers are kept intact and all of the other feature masks are still usable as well. These additions will hopefully provide a more scalable and robust Authorship Attribution system.

References

[1] Narayanan, A., Paskov, H., Gong, N. Z., Bethencourt, J., Stefanov, E., Shin, E. C. R., and Song, D. (2012, May). On the feasibility of internet-scale author identification. Security and Privacy (SP), 2012 IEEE Symposium on (pp. 300-314). IEEE.

[2] C. Faust, G. Dozier, J. Xu, and M. King, Adversarial Authorship, Interactive Evolutionary Hill-Climbing, and AuthorCAAT-III, to appear in: The 2017 IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2017), Honolulu, HI, Nov. 27 Dec. 1, 2017.

[3] Tempestt Neal, Kalaivani Sundararajan, Aneez Fatima, Yiming Yan, Yingfei Xiang, and Damon Woodard. 2017. Surveying Stylometry Techniques and Applications. ACM Comput. Surv. 50, 6, Article 86 (November 2017), 36 pages.

[4] Stamatatos, E. (2009). A survey of modern authorship attribution methods. Journal of the American Society for information Science and Technology,60(3), 538-556.

[5] Schneider, Michael J., ”A Study on the Efficacy of Sentiment Analysis in Author Attribution” (2015). Electronic Theses and Dissertations. Paper 2538.

[6] Panicheva, P., Cardiff, J. and Rosso, P. Personal sense and idiolect: Combining authorship attribution and opinion analysis. Seventh International Conference on Language Resources and Evaluation, Malta, 2010.

[7] Pennebaker, J.W., Boyd, R.L., Jordan, K., and Blackburn, K. (2015). The development and psychometric properties of LIWC2015. Austin, TX: University of Texas at Austin.

[8] Yu, P. L. (1989). Multiple Criteria Decision Making: Five Basic Concepts, Handbooks in Operations Research and Management Science, Vol 1 (Optimization), pp. 663-699, G.L. Nemhauser et al. Eds. Elsevier Science Publishers B.V. (North-Holland).

[9] Davis, L. (1991). Handbook Of Genetic Algorithms. Van Nostrand Reinhold, New York. 115.

[10] Dozier, G., Purrington, K., Popplewell, K., Shelton, J., Bryant, K., Adams, J., Woodard, D. L., and Miller, P. (2011). GEFeS: Genetic and Evolutionary Feature Selection for Periocular Biometric Recognition, Proceedings of the 2011 IEEE Workshop on Computational Intelligence in Biometrics and Identity Management (CIBIM-2011), April 11-15, Paris, France.

[11] T. Baltruaitis, C. Ahuja and L. P. Morency, ”Multimodal Machine Learning: A Survey and Taxonomy,” in IEEE Transactions on Pattern Analysis and Machine Intelligence.

[12] Cees G. M. Snoek, Marcel Worring, and Arnold W. M. Smeulders. 2005. Early versus late fusion in semantic video analysis. In Proceedings of the 13th annual ACM international conference on Multimedia (MULTIMEDIA ’05). ACM, New York, NY, USA, 399-402.

[13] Wiebe, J. Wilson, T., and Cardie, C. (2005). Annotating Expressions of Opinions and Emotions in Language. Language Resourses and Evaluation, 39[2-3], pp. 165-210.

[14] Ellen Riloff, Janyce Wiebe, and Theresa Wilson (2003). Learning Subjective Nouns Using Extraction Pattern Bootstrapping. Seventh Conference on Natural Language Learning (CoNLL-03). ACL SIGNLL.

[15] Ellen Riloff and Janyce Wiebe (2003). Learning Extraction Patterns for Subjective Expressions. Conference on Empirical Methods in Natural Language Processing (EMNLP-03). ACL SIGDAT. Pages 105-112.

[16] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann (2005). Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. Proceedings of Human Language Technologies Conference/Conference on Empirical

[17] Narayanan, M., Gaston, J., Dozier, G., Cothran, D. L., Arms-Chavez, C., Rossi, M., King, M. C, Xu, J. Adversarial Authorship, AuthorWebs, and the AuthorWeb Zoo, To appear in: The 2018 IEEE Symposium Series on Computational Intelligence, Computational Intelligence in Cyber Security Symposium, Special Session: Computational Intelligence in the Identity Ecosystem.

[18] Matwin, S., and Scott, S. (1999). Feature Engineering for Text Classification. Proceedings of the 16th International Conference on Machine Learning, pp. 379-388.

[19] Mack, N., Bowers, J., Williams, H., Dozier, G., Shelton, J. The Best Way to a Strong Defense is a Strong Offense: Mitigating Deanonymization Attacks via Iterative Language Translation. International Journal of Machine Learning and Computing vol.5, no. 5, pp. 409-413, 2015.

[20] Pennebaker J. W., Chung C. K. , Frazee J. , Lavergne G. M., and Beaver D. I. (2014). When small words foretell academic success: The case of college admissions essays. PLoS ONE 9(12): e115844. doi: 10.1371/journal.pone.0115844.

[21] Kacewicz, W., Pennebaker, J.W., Davis, M., Jeon, M., and Graesser, A.C. (2013). Pronoun use reflects standings in social hierarchies. Journal of Language and Social Psychology. online version 19 September 2013, DOI: 10.1177/0261927X1350265.

[22] Newman, M.L., Pennebaker, J.W., Berry, D.S., and Richards, J.M. (2003). Lying words: Predicting deception from linguistic styles. Personality and Social Psychology Bulletin, 29, 665-675.

[23] Pennebaker, J.W. (2011). The Secret Life of Pronouns: What Our Words Say About Us (NY: Bloomsbury).

[24] Cohn, M.A., Mehl, M.R., and Pennebaker, J.W. (2004). Linguistic Markers of Psychological Change Surrounding September 11, 2001. Psychological Science, 15, 687-693.

[25] David M. Blei. 2012. Probabilistic topic models. Commun. ACM 55, 4 (April 2012), 77-84. DOI: https://doi.org/10.1145/2133806.2133826

[26] McCallum, Andrew Kachites. ”MALLET: A Machine Learning for Language Toolkit.” http://mallet.cs.umass.edu. 2002.

[27] Gaston, J., Narayanan, M, Dozier, G., Cothran, D. L., Arms-Chavez, C., Rossi, M., King, M. C., and Xu, J. Authorship Attribution vs. Adversarial Authorship from a LIWC and Sentiment Analysis Perspective, To appear in: The 2018 IEEE Symposium Series on Computational Intelligence, Computational Intelligence in Cyber Security Symposium

[28] Gaston, J., Narayanan, M, Dozier, G., Cothran, D. L., Arms-Chavez, C., Rossi, M., King, M. C., and Xu, J. Authorship Attribution via Evolutionary Hybridization of Sentiment Analysis, LIWC, and Topic Modeling Features, To appear in: The 2018 IEEE Symposium Series on Computational Intelligence, Computational Intelligence in Cyber Security Symposium

[29] Alford, A., Popplewell, K., Dozier, G.V., Bryant, K.S., Kelly, J.C., Adams, J., Abegaz, T., and Shelton, J. (2011). GEFeWS: A Hybrid Genetic-Based Feature Weighting and Selection Algorithm for Multi-Biometric Recognition. MAICS.

Appendices

Appendix A

Sentiment Analysis Feature Consistency

Feature Num	Feature Name	Consistency
33	(StrongSubj,StrongPos) → (StrongSubj,StrongPos)	1
57	(StrongSubj,Neutral) → (StrongSubj,StrongPos)	1
63	(StrongSubj,Neutral) → (WeakSubj,StrongPos)	1
65	(StrongSubj,Neutral) → (WeakSubj,Neutral)	1
136	(WeakSubj,Neutral) → (WeakSubj,WeakPos)	1
150	(WeakSubj,WeakNeg) → (WeakSubj,WeakNeg)	1
88	(StrongSubj,StrongNeg) → (WeakSubj,WeakPos)	0.966667
89	(StrongSubj,StrongNeg) → (WeakSubj,Neutral)	0.966667
112	(WeakSubj,StrongPos) → (WeakSubj,WeakPos)	0.966667
77	(StrongSubj,WeakNeg) → (WeakSubj,Neutral)	0.866667
90	(StrongSubj,StrongNeg) → (WeakSubj,WeakNeg)	0.833333
129	(WeakSubj,Neutral) → (StrongSubj,StrongPos)	0.833333
1	Pr(StrongSubj)	0.8
126	(WeakSubj,WeakPos) → (WeakSubj,WeakNeg)	0.766667
137	(WeakSubj,Neutral) → (WeakSubj,Neutral)	0.766667
160	(WeakSubj,StrongNeg) → (WeakSubj,WeakPos)	0.766667
2	Pr(WeakSubj)	0.7
61	(StrongSubj,Neutral) → (StrongSubj,StrongNeg)	0.7
3	Pr(StrongPos)	0.666667

Feature Num	Feature Name	Consistency
83	(StrongSubj,StrongNeg) → (StrongSubj,Neutral)	0.666667
70	(StrongSubj,WeakNeg) → (StrongSubj,WeakPos)	0.633333
127	(WeakSubj,WeakPos) → (WeakSubj,StrongNeg)	0.633333
105	(WeakSubj,StrongPos) → (StrongSubj,StrongPos)	0.6
139	(WeakSubj,Neutral) → (WeakSubj,StrongNeg)	0.6
10	Pr(StrongSubj—WeakPos)	0.566667
114	(WeakSubj,StrongPos) → (WeakSubj,WeakNeg)	0.566667
17	Pr(WeakSubj—Neutral)	0.533333
29	Pr(Neutral—WeakSubj)	0.533333
4	Pr(WeakPos)	0.466667
36	(StrongSubj,StrongPos) → (StrongSubj,WeakNeg)	0.466667
121	(WeakSubj,WeakPos) → (StrongSubj,StrongNeg)	0.466667
131	(WeakSubj,Neutral) → (StrongSubj,Neutral)	0.466667
39	(StrongSubj,StrongPos) → (WeakSubj,StrongPos)	0.433333
5	Pr(Neutral)	0.4
28	Pr(WeakPos—WeakSubj)	0.4
34	(StrongSubj,StrongPos) → (StrongSubj,WeakPos)	0.4
47	(StrongSubj,WeakPos) → (StrongSubj,Neutral)	0.4
50	(StrongSubj,WeakPos) → (StrongSubj,Both)	0.333333
106	(WeakSubj,StrongPos) → (StrongSubj,WeakPos)	0.333333
123	(WeakSubj,WeakPos) → (WeakSubj,StrongPos)	0.3
143	(WeakSubj,WeakNeg) → (StrongSubj,Neutral)	0.3
21	Pr(StrongPos—StrongSubj)	0.266667
49	(StrongSubj,WeakPos) → (StrongSubj,StrongNeg)	0.266667
73	(StrongSubj,WeakNeg) → (StrongSubj,StrongNeg)	0.266667
117	(WeakSubj,WeakPos) → (StrongSubj,StrongPos)	0.266667
164	(WeakSubj,StrongNeg) → (WeakSubj,Both)	0.233333

Feature Num	Feature Name	Consistency
26	Pr(Both—StrongSubj)	0.2
44	(StrongSubj,StrongPos) → (WeakSubj,Both)	0.2
87	(StrongSubj,StrongNeg) → (WeakSubj,StrongPos)	0.2
100	(StrongSubj,Both) → (WeakSubj,WeakPos)	0.2
104	(StrongSubj,Both) → (WeakSubj,Both)	0.2
141	(WeakSubj,WeakNeg) → (StrongSubj,StrongPos)	0.2
165	(WeakSubj,Both) → (StrongSubj,StrongPos)	0.2
169	(WeakSubj,Both) → (StrongSubj,StrongNeg)	0.2
175	(WeakSubj,Both) → (WeakSubj,StrongNeg)	0.2
9	Pr(StrongSubj—StrongPos)	0.166667
11	Pr(StrongSubj—Neutral)	0.166667
16	Pr(WeakSubj—WeakPos)	0.166667
37	(StrongSubj,StrongPos) → (StrongSubj,StrongNeg)	0.166667
51	(StrongSubj,WeakPos) → (WeakSubj,StrongPos)	0.166667
62	(StrongSubj,Neutral) → (StrongSubj,Both)	0.166667
68	(StrongSubj,Neutral) → (WeakSubj,Both)	0.166667
101	(StrongSubj,Both) → (WeakSubj,Neutral)	0.166667
144	(WeakSubj,WeakNeg) → (StrongSubj,WeakNeg)	0.166667
152	(WeakSubj,WeakNeg) → (WeakSubj,Both)	0.166667
166	(WeakSubj,Both) → (StrongSubj,WeakPos)	0.166667
176	(WeakSubj,Both) → (WeakSubj,Both)	0.166667
23	Pr(Neutral—StrongSubj)	0.133333
43	(StrongSubj,StrongPos) → (WeakSubj,StrongNeg)	0.133333
80	(StrongSubj,WeakNeg) → (WeakSubj,Both)	0.133333
86	(StrongSubj,StrongNeg) → (StrongSubj,Both)	0.133333
93	(StrongSubj,Both) → (StrongSubj,StrongPos)	0.133333
95	(StrongSubj,Both) → (StrongSubj,Neutral)	0.133333

Feature Num	Feature Name	Consistency
102	(StrongSubj,Both) → (WeakSubj,WeakNeg)	0.133333
103	(StrongSubj,Both) → (WeakSubj,StrongNeg)	0.133333
140	(WeakSubj,Neutral) → (WeakSubj,Both)	0.133333
146	(WeakSubj,WeakNeg) → (StrongSubj,Both)	0.133333
159	(WeakSubj,StrongNeg) → (WeakSubj,StrongPos)	0.133333
15	Pr(WeakSubj—StrongPos)	0.1
27	Pr(StrongPos—WeakSubj)	0.1
41	(StrongSubj,StrongPos) → (WeakSubj,Neutral)	0.1
74	(StrongSubj,WeakNeg) → (StrongSubj,Both)	0.1
92	(StrongSubj,StrongNeg) → (WeakSubj,Both)	0.1
94	(StrongSubj,Both) → (StrongSubj,WeakPos)	0.1
96	(StrongSubj,Both) → (StrongSubj,WeakNeg)	0.1
98	(StrongSubj,Both) → (StrongSubj,Both)	0.1
99	(StrongSubj,Both) → (WeakSubj,StrongPos)	0.1
110	(WeakSubj,StrongPos) → (StrongSubj,Both)	0.1
134	(WeakSubj,Neutral) → (StrongSubj,Both)	0.1
162	(WeakSubj,StrongNeg) → (WeakSubj,WeakNeg)	0.1
168	(WeakSubj,Both) → (StrongSubj,WeakNeg)	0.1
171	(WeakSubj,Both) → (WeakSubj,StrongPos)	0.1
173	(WeakSubj,Both) → (WeakSubj,Neutral)	0.1
6	Pr(WeakNeg)	0.066667
7	Pr(StrongNeg)	0.066667
13	Pr(StrongSubj—StrongNeg)	0.066667
14	Pr(StrongSubj—Both)	0.066667
22	Pr(WeakPos—StrongSubj)	0.066667
25	Pr(StrongNeg—StrongSubj)	0.066667
38	(StrongSubj,StrongPos) → (StrongSubj,Both)	0.066667

Feature Num	Feature Name	Consistency
58	(StrongSubj,Neutral) → (StrongSubj,WeakPos)	0.066667
119	(WeakSubj,WeakPos) → (StrongSubj,Neutral)	0.066667
128	(WeakSubj,WeakPos) → (WeakSubj,Both)	0.066667
135	(WeakSubj,Neutral) → (WeakSubj,StrongPos)	0.066667
145	(WeakSubj,WeakNeg) → (StrongSubj,StrongNeg)	0.066667
158	(WeakSubj,StrongNeg) → (StrongSubj,Both)	0.066667
170	(WeakSubj,Both) → (StrongSubj,Both)	0.066667
172	(WeakSubj,Both) → (WeakSubj,WeakPos)	0.066667
20	Pr(WeakSubj—Both)	0.033333
30	Pr(WeakNeg—WeakSubj)	0.033333
32	Pr(Both—WeakSubj)	0.033333
35	(StrongSubj,StrongPos) → (StrongSubj,Neutral)	0.033333
60	(StrongSubj,Neutral) → (StrongSubj,WeakNeg)	0.033333
66	(StrongSubj,Neutral) → (WeakSubj,WeakNeg)	0.033333
69	(StrongSubj,WeakNeg) → (StrongSubj,StrongPos)	0.033333
71	(StrongSubj,WeakNeg) → (StrongSubj,Neutral)	0.033333
85	(StrongSubj,StrongNeg) → (StrongSubj,StrongNeg)	0.033333
91	(StrongSubj,StrongNeg) → (WeakSubj,StrongNeg)	0.033333
97	(StrongSubj,Both) → (StrongSubj,StrongNeg)	0.033333
111	(WeakSubj,StrongPos) → (WeakSubj,StrongPos)	0.033333
116	(WeakSubj,StrongPos) → (WeakSubj,Both)	0.033333
125	(WeakSubj,WeakPos) → (WeakSubj,Neutral)	0.033333
132	(WeakSubj,Neutral) → (StrongSubj,WeakNeg)	0.033333
148	(WeakSubj,WeakNeg) → (WeakSubj,WeakPos)	0.033333
149	(WeakSubj,WeakNeg) → (WeakSubj,Neutral)	0.033333
157	(WeakSubj,StrongNeg) → (StrongSubj,StrongNeg)	0.033333
163	(WeakSubj,StrongNeg) → (WeakSubj,StrongNeg)	0.033333

Feature Num	Feature Name	Consistency
167	(WeakSubj,Both) → (StrongSubj,Neutral)	0.033333
174	(WeakSubj,Both) → (WeakSubj,WeakNeg)	0.033333
8	Pr(Both)	0
12	Pr(StrongSubj—WeakNeg)	0
18	Pr(WeakSubj—WeakNeg)	0
19	Pr(WeakSubj—StrongNeg)	0
24	Pr(WeakNeg—StrongSubj)	0
31	Pr(StrongNeg—WeakSubj)	0
40	(StrongSubj,StrongPos) → (WeakSubj,WeakPos)	0
42	(StrongSubj,StrongPos) → (WeakSubj,WeakNeg)	0
45	(StrongSubj,WeakPos) → (StrongSubj,StrongPos)	0
46	(StrongSubj,WeakPos) → (StrongSubj,WeakPos)	0
48	(StrongSubj,WeakPos) → (StrongSubj,WeakNeg)	0
52	(StrongSubj,WeakPos) → (WeakSubj,WeakPos)	0
53	(StrongSubj,WeakPos) → (WeakSubj,Neutral)	0
54	(StrongSubj,WeakPos) → (WeakSubj,WeakNeg)	0
55	(StrongSubj,WeakPos) → (WeakSubj,StrongNeg)	0
56	(StrongSubj,WeakPos) → (WeakSubj,Both)	0
59	(StrongSubj,Neutral) → (StrongSubj,Neutral)	0
64	(StrongSubj,Neutral) → (WeakSubj,WeakPos)	0
67	(StrongSubj,Neutral) → (WeakSubj,StrongNeg)	0
72	(StrongSubj,WeakNeg) → (StrongSubj,WeakNeg)	0
75	(StrongSubj,WeakNeg) → (WeakSubj,StrongPos)	0
76	(StrongSubj,WeakNeg) → (WeakSubj,WeakPos)	0
78	(StrongSubj,WeakNeg) → (WeakSubj,WeakNeg)	0
79	(StrongSubj,WeakNeg) → (WeakSubj,StrongNeg)	0
81	(StrongSubj,StrongNeg) → (StrongSubj,StrongPos)	0

Feature Num	Feature Name	Consistency
82	(StrongSubj,StrongNeg) → (StrongSubj,WeakPos)	0
84	(StrongSubj,StrongNeg) → (StrongSubj,WeakNeg)	0
107	(WeakSubj,StrongPos) → (StrongSubj,Neutral)	0
108	(WeakSubj,StrongPos) → (StrongSubj,WeakNeg)	0
109	(WeakSubj,StrongPos) → (StrongSubj,StrongNeg)	0
113	(WeakSubj,StrongPos) → (WeakSubj,Neutral)	0
115	(WeakSubj,StrongPos) → (WeakSubj,StrongNeg)	0
118	(WeakSubj,WeakPos) → (StrongSubj,WeakPos)	0
120	(WeakSubj,WeakPos) → (StrongSubj,WeakNeg)	0
122	(WeakSubj,WeakPos) → (StrongSubj,Both)	0
124	(WeakSubj,WeakPos) → (WeakSubj,WeakPos)	0
130	(WeakSubj,Neutral) → (StrongSubj,WeakPos)	0
133	(WeakSubj,Neutral) → (StrongSubj,StrongNeg)	0
138	(WeakSubj,Neutral) → (WeakSubj,WeakNeg)	0
142	(WeakSubj,WeakNeg) → (StrongSubj,WeakPos)	0
147	(WeakSubj,WeakNeg) → (WeakSubj,StrongPos)	0
151	(WeakSubj,WeakNeg) → (WeakSubj,StrongNeg)	0
153	(WeakSubj,StrongNeg) → (StrongSubj,StrongPos)	0
154	(WeakSubj,StrongNeg) → (StrongSubj,WeakPos)	0
155	(WeakSubj,StrongNeg) → (StrongSubj,Neutral)	0
156	(WeakSubj,StrongNeg) → (StrongSubj,WeakNeg)	0
161	(WeakSubj,StrongNeg) → (WeakSubj,Neutral)	0

Appendix B

LIWC Feature Consistency

Feature Number	Feature Name	Consistency
1	WC	1
6	WPS	1
7	Sixltr	1
39	female	1
42	insight	1
43	cause	1
60	power	1
70	work	1
73	money	1
83	Period	1
89	Dash	1
91	Apostro	1
24	verb	0.933333
49	see	0.933333
54	health	0.933333
19	prep	0.9
85	Colon	0.9
63	focuspast	0.866667
87	Qmark	0.833333

Feature Number	Feature Name	Consistency
21	adverb	0.8
71	leisure	0.8
74	relig	0.766667
44	discrep	0.733333
12	i	0.633333
59	achieve	0.633333
66	relativ	0.633333
40	male	0.6
15	shehe	0.566667
52	bio	0.566667
9	function	0.466667
50	hear	0.466667
51	feel	0.433333
78	netspeak	0.433333
31	posemo	0.4
32	negemo	0.4
33	anx	0.4
56	ingest	0.4
5	Tone	0.366667
67	motion	0.366667
82	AllPunc	0.366667
22	conj	0.333333
11	ppron	0.3
30	affect	0.3
4	Authentic	0.233333
58	affiliation	0.233333
64	focuspresent	0.233333

Feature Number	Feature Name	Consistency
14	you	0.133333
45	tentat	0.133333
90	Quote	0.133333
62	risk	0.1
20	auxverb	0.066667
41	cogproc	0.066667
48	percept	0.066667
10	pronoun	0.033333
18	article	0.033333
23	negate	0.033333
28	number	0.033333
29	quant	0.033333
35	sad	0.033333
36	social	0.033333
61	reward	0.033333
68	space	0.033333
72	home	0.033333
88	Exclam	0.033333
93	OtherP	0.033333
2	Analytic	0
3	Clout	0
8	Dic	0
13	we	0
16	they	0
17	ipron	0
25	adj	0
26	compare	0

Feature Number	Feature Name	Consistency
27	interrog	0
34	anger	0
37	family	0
38	friend	0
46	certain	0
47	differ	0
53	body	0
55	sexual	0
57	drives	0
65	focusfuture	0
69	time	0
75	death	0
76	informal	0
77	swear	0
79	assent	0
80	nonflu	0
81	filler	0
84	Comma	0
86	SemiC	0
92	Parenth	0

Appendix C

Topic Model Feature Consistency

Topic Number	Consistency
1	1
2	1
4	1
5	1
6	1
8	1
10	1
11	1
12	1
13	1
14	1
16	1
17	1
18	1
21	1
22	1
24	1
26	1
27	1

Topic Number	Consistency
29	1
30	1
31	1
32	1
34	1
36	1
37	1
38	1
45	1
25	0.733333
43	0.266667
3	0
7	0
9	0
15	0
19	0
20	0
23	0
28	0
33	0
35	0
39	0
40	0
41	0
42	0
44	0