An Exploration of Methods for Identifying Conversation Participants

Alexicia Richardson
Department of Computer Science and Software Engineering
Auburn University
Auburn, USA
adr0021@auburn.edu

Manik Thogaripally
Department of Computer Science and Software Engineering
Auburn University
Auburn, USA
mrt0037@auburn.edu

Sarp Aykent
Department of Computer Science and Software Engineering
Auburn University
Auburn, USA
sza0112@auburn.edu

Richard Chapman
Department of Computer Science and Software Engineering
Auburn University
Auburn, USA
chapmro@auburn.edu

Gerry Dozier
Department of Computer Science and Software Engineering
Auburn University
Auburn, USA
doziegv@auburn.edu

***Abstract*—In this paper, we investigate the use of a number of methods for identifying the participants of conversations. These methods include: Twitchell's Speech Acts Profiling, five well-known authorship attribution methods, a technique that has been used to identify co-authors of collaborative written text, a neural network and a deep neural network. Our results suggest that for written text, most of the methods perform well. However, they perform poorly on transcripts of spoken dialogue, such as the Friends sitcom transcripts and chat text. Our results show that as the number of conversation turns increases, so does the accuracy of identifying conversation participants. However, this comes at a cost of a reduced number of conversations within the dataset.**

I. Introduction

The area of authorship analysis [18] has given rise to a number of authorship attribution methods for identifying the authors of text. Currently, there are a number of researchers that are using many of these authorship attribution methods in an effort to identify the co-authors of collaboratively written text documents. Documents of text-based conversations [1][2][3], can be viewed as collaboratively written texts by a group of co-authors. In this paper, we explore the use of authorship attribution methods for identifying participants of conversations on a dataset that was developed using scripts from the TV sitcom, Friends [19], the Multi-Party Chat Corpus, MPC, dataset [25], and the C50 dataset [26].

The remainder of this paper is as follows. In Section II, we briefly present some related work, in Section III, a number of methods that can be potentially used for identifying the participants of a conversation are introduced. In Section IV, our experiment is presented along with our datasets and in Section V, we present the results of our experiment. In Section VI, we provide our conclusions and future work.

II. Related Work

In [10], Dauber et al. study a number of approaches for identifying the authors of collaboratively written text. In their study, they developed two experiments:

1. train on text of single authors and test on collaboratively written text and
2. train and test exclusively on collaboratively written text.

For both experiments, Dauber et al. developed two baselines for determining the accuracy of the Writeprints (Limited) Sequential Minimal Optimization Linear Support Vector Machine (SMO-Writeprints$_{\text{LTD}}$) [11]. The first baseline consisted of ten authors and had an accuracy of 51.3%. The second consisted of 75 authors and had an accuracy of 14.2%.

In the first experiment by Dauber et al., they train on text written by a single author and test on collaboratively written text. The training set of the experiment consisted of two variations:

1. having known text for each author and
2. having known text for some of the authors.

Each variation of the training set was used to evaluate the test set of unknown collaboratively written text. The first variation enables Dauber et al. to potentially identify all authors of the text. The second variation restricts Dauber et al. to only being able to make an identification based on the authors they have text for. For each variation, Dauber et al. attempts to identify the authors of text written collaboratively by two, three, and four authors.

In an effort to measure accuracy of the experiment, a variety of metrics were used:

a. LSVM example-based accuracy (EBA),
b. LSVM guess-one (GO),
c. Label Powerset LSVM example-based accuracy (LPEBA),
d. LSVM Subset (LS), and
e. Label Powerset LSVM Subset (LPLS).

When using the first variation of the training set to evaluate the test set in an effort to identify two, three, and four authors, only three of the five metrics were used, EBA, GO, and LPEBA. When measuring the accuracy of identifying two authors, EBA

978-1-7281-6861-6/20/$31.00 ©2020 IEEE

had an accuracy of approximately 22%, GO had an accuracy of approximately 45%, and LPEBA had an accuracy of approximately 24%. Using the metrics to measure the accuracy of identifying three authors, EBA had an accuracy of about 13%, GO had an accuracy of about 37%, and LPEBA had an accuracy of about 15%. For identifying four authors of an unknown text, EBA attained an accuracy of roughly 9%, GO roughly 27%, and LPEBA roughly 8%.

The second variation of the training set uses all metrics listed to measure accuracy of identifying two, three, and four authors. For two authors, EBA, GO, and LS had an accuracy of approximately 23% while LPEBA and LPLS had an accuracy of approximately 20%. For three authors, EBA had an accuracy of about 16%, GO had an accuracy of about 19%, LS and LPLS had an accuracy of about 12%, and LPEBA had an accuracy of about 17%. For four authors, EBA and LPEBA had an accuracy of roughly 15%, GO had an accuracy of roughly 19%, LS had an accuracy of roughly 10%, and LPLS had an accuracy of roughly 9.5%.

In the second experiment studied by Dauber et al., they used collaboratively written text for both the training and test set. The experiment had three variations:

1. train and test on collaborative text of groups of co-authors,
2. train and test on set of collaborative text where some groups had not worked together, and
3. train and test on collaborative text where the groups have not worked together.

For the first variation of the experiment, the following metrics were used to identify two, three, and four authors:

a. Comparable Single Author Baseline Split (CSABS),
b. Comparable Single Author Baseline Unit (CSABU),
c. LSVM Unit Accuracy (UA),
d. Label Powerset SVM Split example-based accuracy (SEBA),
e. Label Powerset LSVM Split Subset (SS), and
f. Label Powerset SVM Unit Accuracy (LPUA)

When identifying two authors, CSABS and CSABU achieved an accuracy of approximately 15%, UA of approximately 39%, SEBA of approximately 47%, SS and LPUA of approximately 38%. For three authors, CSABS achieved an accuracy of about 19%, CSABU of about 35%, UA of about 82%, SEBA, SS, LPUA of about 88%. For four authors, CSABS achieved an accuracy of roughly 28%, CSABU of roughly 50%, UA, SEBA, SS, and LPUA of roughly 100%.

For the next two variations of the experiment, the following metrics were used to identify from two to seven authors:

a. EBA
b. LS
c. LPEBA
d. LPLS
e. Binary Relevance Naïve Bayes example-based accuracy (EBA_{BRNB})

The second variation of the experiment achieves an accuracy of approximately 25% when using EBA to identify two authors, approximately 10% for LS, approximately 16% for LPEBA, and approximately 6% for LPLS. For three authors, EBA has an accuracy of about 18%, LS about 6%, LPEBA about 14%, and about 8% for LPLS. For four authors, using EBA achieved an accuracy of roughly 15%, LS roughly 5%, LPEBA roughly 12%, and LPLS roughly 6%. In the case of five authors, EBA had an accuracy of around 15%, LS around 3%, LPEBA around 12%, and LPLS around 5%. For six authors, EBA had an accuracy of approximately 15%, LS and LPLS approximately 5%, and LPEBA approximately 11%. For seven authors, EBA had an accuracy of about 14%, LPEBA of about 10%, and LPLS of about 4%.

The third variation of the experiment uses only two metrics to identify between two and seven authors of unknown text. For two authors, using EBA_{BRNB}, the accuracy was about 10% and about 15% for EBA. Using EBA_{BRNB} for three authors has an accuracy of about 9% and EBA about 12%. In the case of four authors, EBA_{BRNB} achieved an accuracy of around 8% and EBA around 12%. For five authors, EBA_{BRNB} had an accuracy of roughly 10%, while EBA had an accuracy of roughly 13%. When it comes to six authors, EBA_{BRNB} had an accuracy of around 10% and EBA of around 12%. For seven authors, EBA_{BRNB} had an accuracy of approximately 12%, and EBA approximately 11%.

The results in this paper are based on an experiment that is most similar to the first variation of the second experiment studied by Dauber et al.: training and testing on collaboratively written text.

In [23], Sari et al. uses a neural network for their continuous n-gram feature representation in an effort to identify the author of a given text. Their network has three variations:

1. word unigrams and bigrams,
2. character unigrams, bigrams, trigrams, and four-grams, and
3. a combination of 1 and 2.

They test their algorithm on four datasets: a Judgment dataset, the CCAT10 and CCAT50 dataset, and the IMDB62 dataset. For the Judgment and IMDb62 dataset, they used cross validation, splitting the datasets into 10-folds, while for CCAT10 and CCAT50 they split the dataset 50-50 for a train and test set. For each dataset, the network model consisted of an embedding layer, an average pooling 1D layer, a flatten layer, and a dense layer. For the implementation of the word-char combination, they merged two models containing those layers together, one representing characters and the other words.

Sari et al. tested their networks on the Judgment dataset and with the word unigram-bigram variation achieved an accuracy of 90.31%, the character combination had an accuracy of 91.29%, and the concatenation of the two achieved an accuracy of 91.51%. On the CCAT10 dataset, the word variation had an accuracy of 77.80%, the character variation 74.80%, and the combination of the two had an accuracy of 77.20%. For the CCAT50 dataset, the word model had an accuracy of 70.16%, while the character model achieved an accuracy of 72.60%, and the model representing the combination of the two had an accuracy of 72.04%. The word model had an accuracy of 87.87% on the IMDb62 dataset, while the character model had an accuracy of 94.80%, and the combination of the two achieved an accuracy of 94.28%.

In [24], Shrestha et al. uses a convolutional neural network in an effort to identify the authors of short text. The CNN consisted of an embedding layer, a dropout layer, a convolutional layer, and a fully connected layer. Shrestha et al. uses a sequence of character unigrams as one variation and another using a sequence of character bigrams. Within the code, there also exists a word n-gram option, which we tested our datasets on using word unigrams.

Shrestha et al. test the effectiveness of their algorithm on a twitter dataset. Given 50 authors, each having 1000 tweets, their character unigram CNN had an accuracy of 75.7% while their character bigram CNN had an accuracy of 76.1%.

To test the effectiveness of the author count on their models, they experimented with different numbers of authors all having 200 tweets each. For 100 authors, their CNN using character unigrams had an accuracy of 50.8% and its counterpart using bigrams had an accuracy of 50.6%. In the case of 200 authors, the unigram CNN achieved an accuracy of 47.3%, while the bigram CNN obtained an accuracy of 48.1%. For 500 authors, the unigram CNN had an accuracy of 41.7% and the bigram had an accuracy of 42.2%. In the case of 1000 authors, the first variation using unigrams had an accuracy of 35.9% and bigrams 36.5%.

In an effort to see how the tweet count effects the results, they used the 50 authors and altered the number of tweets per author. With each author having 500 tweets, the unigram CNN had an accuracy of 71.7%, while the bigram's accuracy was 72.4%. Reducing the tweet count to 200 leads to the unigram and bigram variation of the CNN's accuracy dropping to 66.5%. When the tweet count drops to 100, the unigram CNN had an accuracy of 61.7%, and the bigram CNN had an accuracy of 61.3%. Given 50 tweets per authors, the unigram CNN had an accuracy of 56.2%, while the bigram variation had an accuracy of 54.2%.

Another experiment by Shrestha et al. involved analyzing only the responses that were made by humans, which meant getting rid of any authors that tweets appeared to be machine produced. With their author set being reduced to 35 authors, each with 1000 tweets, the unigram-based CNN had an accuracy of 67.8% while the bigram-based CNN had an accuracy of 68.3%.

III. Methods for Identifying Conversation Participants

In this section, we present a number of methods that can potentially be used to identify conversation participants. These methods can be viewed as belonging to two classes: speech act profiling [1][2][3] and authorship attribution [18].

A. *Speech Act Profiling*

For Twitchell's Speech Act Tagging, a feature vector representing the frequency of 42 SWBD-DAMSL dialogue acts is created and used for Speech Act Profiling [4][20][21]. For the case of identifying conversation participants, a feature vector of 84 speech acts is developed. The first 42 speech acts are mapped to the first individual of the conversation and the last 42 are mapped to the second individual.

B. *Authorship Attribution*

In an effort to apply authorship attribution for identifying conversation participants, one can concatenate a conversation into one writing sample and label that writing sample using the author pair. In what follows, is a brief description of five well known authorship attribution systems that were explored.

Keselj et al. [5], implements a byte-level n-gram profile for each author. The differences between the author profiles are calculated through the Profile Dissimilarity algorithm, which can quantify the dissimilarity between writing styles. The text is classified to a known author class with the minimal dissimilarity calculated. This method has been shown to be effective on English data sets with large data sizes and a small (7) set of authors.

Benedetto et al. [6], attempts to solve the problem of authorship attribution by calculating the relative entropy between texts. In order to calculate the relative entropy, the LZ77 compression [22] is used. For example, if α and β are texts and *b* is a subsequence of β, then the relative entropy of α and β is the entropy of α+*b* minus the entropy of β+*b*. The method of appending a subsequence to the end ensures that if *b* is different from α then the compression would be suboptimal. Benedetto's method has been tested on language recognition, authorship attribution, and the classification of sequences.

Stamatatos [7] uses several single linear discriminant classifiers trained on a varying number of word-tokens, for authorship attribution. The final decision of the ensemble classifier is made by taking the average of two methods, the product and mean. This method has been tested on newspaper articles in Greek. Stamatatos' ensemble classifier is effective if the text size is large (800+ words); however, its performance is slightly less than Keselj et al. approach [5] in authorship attribution challenges in Greek [7].

Koppel et al. [8] demonstrated that space-free character 4-gram features are effective in English and Greek authorship attribution. In addition to attributing a text to a particular author, they allow for the algorithm to classify a text as 'Don't Know.'

The method proposed by Koppel et al. compares random subsequences and finds the closest match in a known text. Then the relative score is calculated and compared with a threshold. If the score is larger than the threshold a classification is made. On the other hand, if the score is lower than the threshold, the method will classify the text as 'Don't Know,' to represent the fact that the text in question was not authored by any of the set of known authors.

Benedetto et al. [6] and Teahan and Harper [9] use a compression-based model for authorship attribution. Teahan relies on Prediction by Partial Matching (PPM) text compression technique as opposed to LZ77. A character-based feature selection is used, and optimization is done on a category-by-category basis. The proposed method of Teahan and Harper was shown to have an accuracy of 93% in authorship identification from a set of 11 possible authors.

IV. EXPERIMENTS

There were three datasets used for our experiments. The first dataset used for our experiment was taken from scripts of the TV sitcom, Friends. The scripts were parsed to discover conversations between two actors. Conversations between any two actors consisted of two turns [12, 13] for each actor. A turn consists of an utterance of one actor. The authors of the conversation were limited to the 6 lead actors: Rachel, Joey, Phoebe, Chandler, Ross, Monica. Our dataset consisted of 5,892 conversations. The second dataset was the Multi-Party Chat Corpus, MPC, developed by Shaikh et al [25]. This dataset consisted of 14 chat sessions that we split into two turn conversations. The authors of the conversation were restricted to the conversation pairs that had at least the floor of the average number of conversations. Our MPC dataset consisted of 663 conversations. The third dataset was the C50 dataset [26]. This dataset consists of 50 authors, each having 100 writing samples. The writing samples were combined to create a the C25 dataset. The C25 dataset consisted of the combinations of every two authors' writing samples and consisted of 2500 samples.

For a baseline, we identified each individual author based on their two turns taken per conversation. The baseline for the Friends dataset consisted of 11,784 samples, the MPC dataset 1,326 samples, and C50, 5000 samples.

For a third experiment, we took the top 5 performing algorithms on the Friends conversation dataset to further evaluate their performance. We had 11 variations of the Friends conversation dataset where each dataset represented a different range of turns going from 2 to 12. In order for an author pair to be used in evaluation, they needed at least 10 conversations. For the 2-turn dataset, we have 5,892 conversations between 15 author pairs. The 3 turn dataset has 2,700 conversations between 15 authors pairs. The 4-turn dataset has 1,560 conversations between 15 author pairs, while 5 turns has 1,010 conversations between 15 author pairs, and 6 turns has 654 conversations between 15 author pairs. The 7-turn dataset author pairs reduce to 12 and have a total of 449 conversations. For 8 turns, there are 319 conversations between 11 author pairs. In the case of 9 turns, the author pair amount is down to 9 and have a total of 214 conversations. For 10 turns, there are 138 conversations between 6 author pairs. There are 87 conversations between 5 authors pairs for 11 turns. The 12-turn dataset has 47 conversations between 3 author pairs.

For the fourth experiment, we took the top 5 performing algorithms on the MPC conversation dataset and tested to see how well they performed on the 2 and 3 turn variations of the dataset. In order for an author pair to be used in this experiment, they needed at least 10 conversations. The 2-turn dataset had 405 conversations between 25 author pairs, while the 3-turn had 38 conversations between 3 author pairs.

For these experiments, as we increase the number of turns, the balance of the datasets will shift. The MPC and Friends datasets were initially unbalanced, while the C50 and C25 datasets were balanced.

V. RESULTS

The results presented in this section were generated as follows. For Speech Act Profiling, we used a Single-Tagged, and a Multi-Tagged approach. In the Single-Tagged approach, each tag can only exist a maximum of once in a turn. In the Multi-Tagged approach, each tag can exist more than once in a turn. For both Speech Act Profiling approaches, we created a linear support vector machine, using the scikit-learn python package with its default parameters. The linear support vector machine took in the 84-length feature vector and outputted an output vector whose length corresponded to the number of authors. In addition, we developed a multi-layer perceptron (MLP) that consists of an input layer of 84 corresponding to our feature vector followed by two hidden layers consisting of 1000 neurons, followed by an output layer whose length corresponded to the potential participants. For the hidden layer, the neurons used the ReLU [14] [15] [16] [17] activation function. The MLP ran for 10,000 epochs with a learning rate of 0.001, using 4-fold cross validation.

For the authorship attribution systems except Keselj2003, we used leave-one-out training on all conversations and the systems were used to classify the conversation as one of the possible combinations of actors. For Keselj2003, we used an 80-20 split train and validation sets. We used the validation set to select the parameters that gave us the best accuracy on the validation set. Those same parameters are used to classify the test set samples. For the test set of Keselj2003, we used leave one out, taking one sample at a time from the training set for testing. For SMO-Writeprints$_{\text{Ltd}}$, each dataset was classified using 5-fold cross validation except the MPC conversations due to lack of data it used 4-fold. In order to be classified using Sari2017 and Shrestha2017, the datasets were split into train and test sets. All instances of the datasets were split into 90% training set and 10% test set. For Sari2017 and Shrestha2017, 10% of the training set is used for validation. For Shrestha2017, the validation set is comprised of 20% of the training set for MPC conversations. Also, while using Shrestha's method, we changed the batch size from 32 to 16 to avoid a memory leak.

Our baseline results are shown in Table I. Table I consists of four columns. The first column represents the methods that are being compared. The second column represents the C50 dataset, while the second column represents the Friends dataset, and the third column represents the MPC dataset. The fourth column represents the associated accuracies for the methods in the first column. In Table I, one can see that Teahan2003 has the best performance on the C50 and MPC dataset, while Shrestha2017$_{\text{word-unigram}}$ performs best on Friends. On all datasets, both the Single-Tagged and Multi-Tagged approaches, the MLPs had better performances than the LSVMs. On the C50 and Friends dataset, neither LSVM nor MLP variation outperformed any of the authorship attribution methods. For the MPC dataset, both LSVM (Single-Tagged and Multi-Tagged) were only able to outperform Keselj2003. The MLP (Single-Tagged and MultiTagged) outperformed Koppel2011, Stamatatos2006 and Keselj2003.

For Experiment 3, the results are displayed in Figures 1 – 5. In Figure 1, the highest accuracy occurred where the conversations had 7 turns and the lowest happens where there are 10 turns. In Figure 2, the highest accuracy occurred at 10 turns where the lowest occurred at 11 turns. For Figures 3 - 5, the highest accuracy is attained at 12 turns and the lowest at 5 for Figure 3 and 6 turns for Figures 4 and 5. The highest accuracy of all was achieved in Figure 2 using Shrestha's character bigrams on 10 turns.

Figures 6 through 10 represent the results from experiment 4. In all cases, they performed better on 3 turns than 2 turns. The highest accuracy came using Shrestha's word unigrams in Figure 10.

The results of our experiment are shown in Table II. As with Table I, Table II consists of four columns representing the methods, datasets, and the associated accuracy. In Table II, one can see that the best performing method on C25 and MPC is Teahan2003, while Shrestha2017$_{\text{word-unigram}}$ is best on Friends. As in our baseline, the MLPs had better on than their LSVM counterparts. For C25, the LSVMs and MLPs were not able to outperform any of the authorship attribution methods. On the Friends dataset, the LSVMs and MLPs only outperformed Koppel2011. On the MPC dataset, the LSVM Multi-Tagged outperformed Keselj2003, while the Single-Tagged did not outperform any method. The MLPs were able to outperform Keselj2003, Stamatatos2006, Sari2017$_{\text{char}}$, Sari2017$_{\text{word}}$, and the Multi-tagged approach also outperformed Shrestha2017$_{\text{char-bigram}}$.

TABLE I. A COMPARISON OF METHODS FOR IDENTIFIYING INDIVIDUAL PARTICIPANTS

	Results		
Method	**C50**	**Friends**	**MPC**
Keselj2003 [5]	81.80%	28.80%	1.4%
Teahan2003 [9]	**99.84%**	30.47%	**37.10%**
SMO-Writeprints$_{\text{Ltd}}$ [11]	78.04%	27.82%	21.57%
Benedetto2002 [6]	84.30%	25.83%	17.50%
Stamatatos2006 [7]	30.30%	19.55%	13.20%
Koppel2011 [8]	67.70%	19.13%	13.73%
LSVM (Single-Tagged)	2.16%	16.62%	5.66%
MLP (Single-Tagged)	9.46%	20.77%	13.80%
LSVM (Multi-Tagged)	1.98%	16.78%	6.33%
MLP (Multi-Tagged)	11.48%	20.62%	15.91%
Sari2017$_{\text{char}}$ [23]	89.00%	23.79%	20.27%
Sari2017$_{\text{word}}$[23]	86.4%	30.82%	23.65%
Sari2017$_{\text{wordchar}}$[23]	89.00%	31.92%	18.24%
Shrestha2017$_{\text{char-unigram}}$[24]	83.80%	33.11%	30.41%
Shrestha2017$_{\text{char-bigram}}$[24]	86.80%	32.85%	36.49%
Shrestha2017$_{\text{word-unigram}}$[24]	83.80%	**33.28%**	31.76%

TABLE II. A COMPARISON OF METHODS FOR IDENTIFIYING CONVERSATION PARTICIPANTS

	Results		
Method	**C25**	**Friends**	**MPC**
Keselj2003 [5]	96.70%	25.80%	2.8%
Teahan2003 [9]	**98.44%**	28.41%	**24.74%**
SMO-Writeprints$_{\text{Ltd}}$ [11]	91.64%	19.45%	12.22%
Benedetto2002 [6]	94.48%	18.23%	16.59%
Stamatatos2006 [7]	41.44%	15.38%	6.18%
Koppel2011 [8]	87.56%	10.79%	11.16%
LSVM (Single-Tagged)	4.00%	11.96%	1.81%
MLP (Single-Tagged)	16.6%	12.98%	8.45%
LSVM (Multi-Tagged)	4.48%	11.40%	3.32%
MLP (Multi-Tagged)	23.4%	13.05%	8.90%
Sari2017$_{\text{char}}$ [23]	91.20%	30.13%	7.77%
Sari2017$_{\text{word}}$[23]	95.60%	29.46%	7.77%
Sari2017$_{\text{wordchar}}$[23]	96.00%	28.62%	6.80%
Shrestha2017$_{\text{char-unigram}}$[24]	96.00%	27.10%	8.74%
Shrestha2017$_{\text{char-bigram}}$[24]	98.40%	29.80%	13.59%
Shrestha2017$_{\text{word-unigram}}$[24]	96.00%	**30.81%**	12.62%

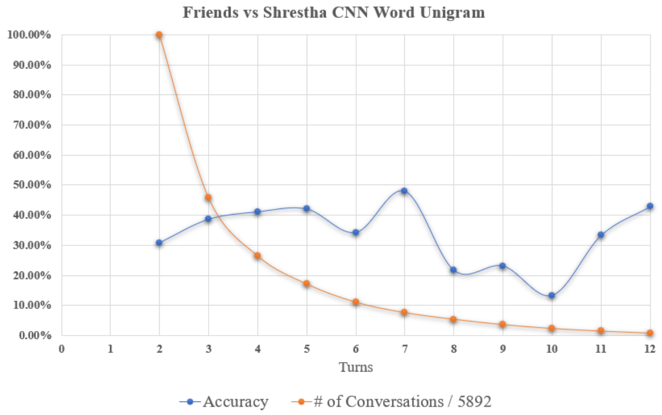

Fig. 1. Friends vs .Shrestha's word unigram (accuracy vs conversation)

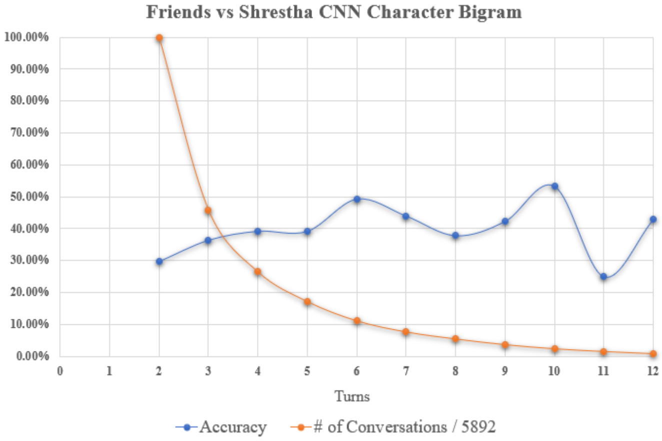

Fig. 2. Friends vs Shrestha's character bigram (accuracy vs conversation)

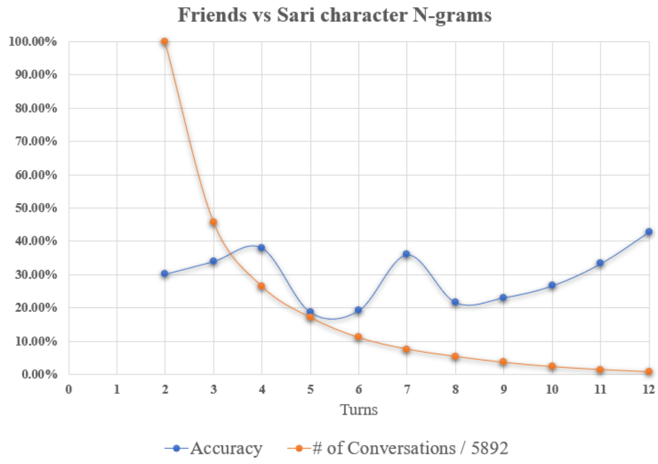

Fig. 3. Friends vs Sari character N-grams (accuracy vs conversation)

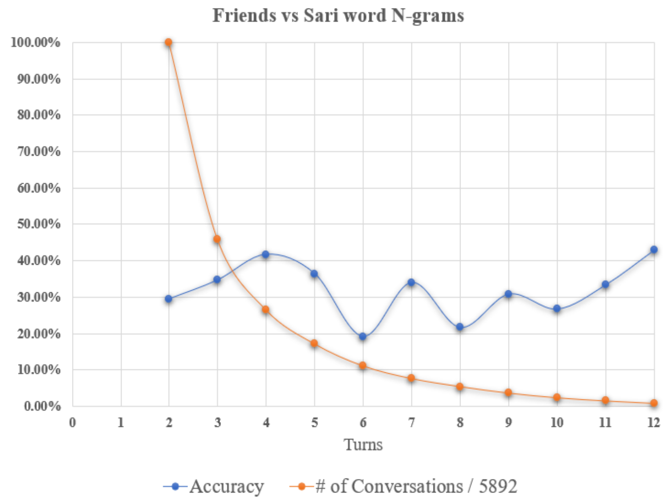

Fig. 4. Friends vs Sari word N-grams (accuracy vs conversation)

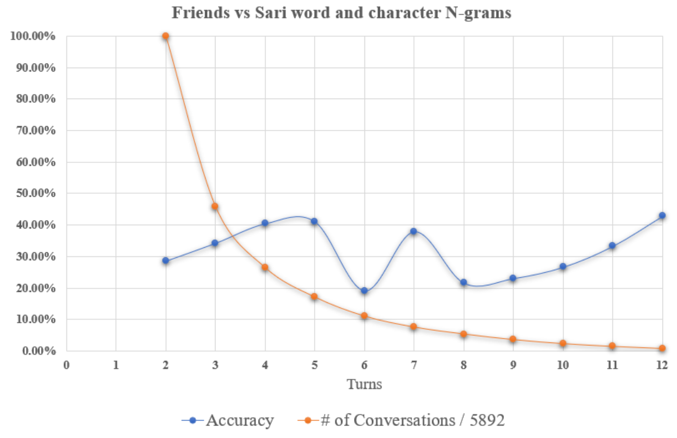

Fig. 5. Friends vs Sari word and character N-grams (accuracy vs conversation)

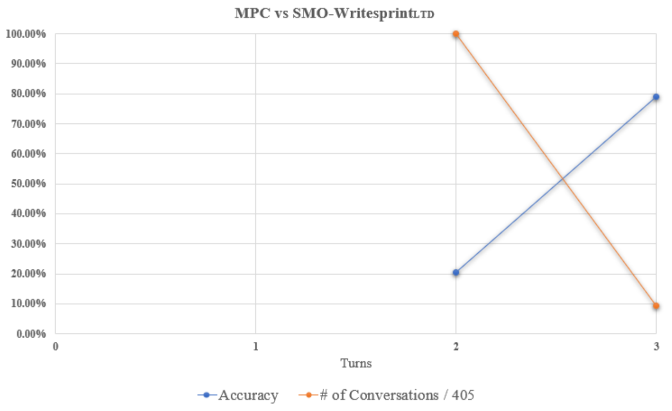

Fig. 6. MPC vs SMO-Writesprint$_{\text{LTD}}$ (accuracy vs conversation)

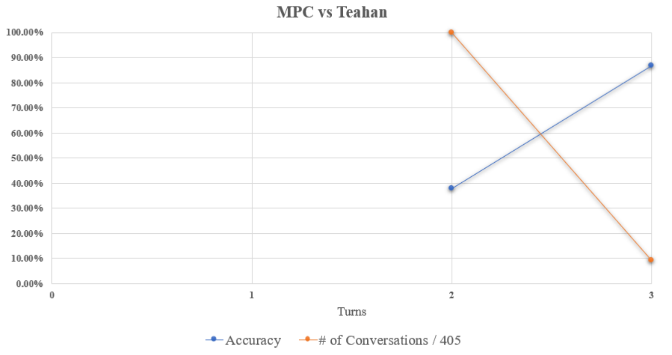

Fig. 7. MPC vs Teahan (accuracy vs conversation)

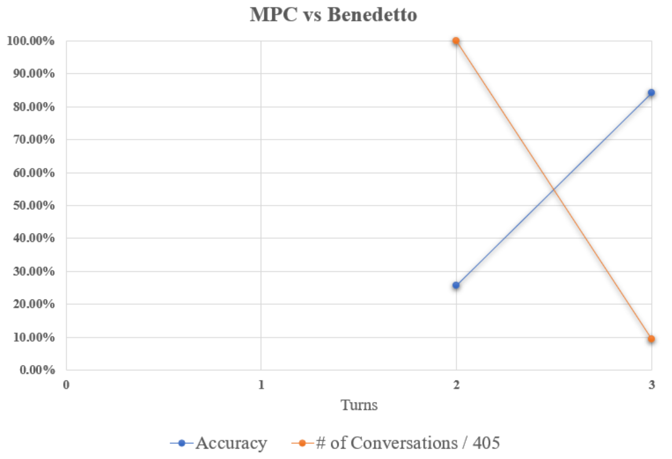

Fig. 8. MPC vs Benedetto (accuracy vs conversation)

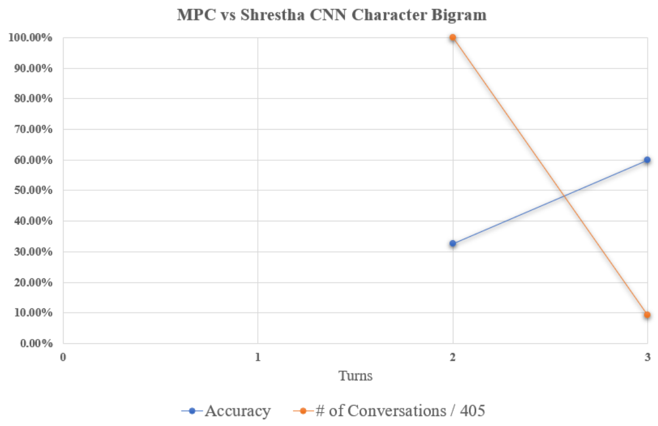

Fig. 9. MPC vs Shrestha’s character bigram (accuracy vs conversation)

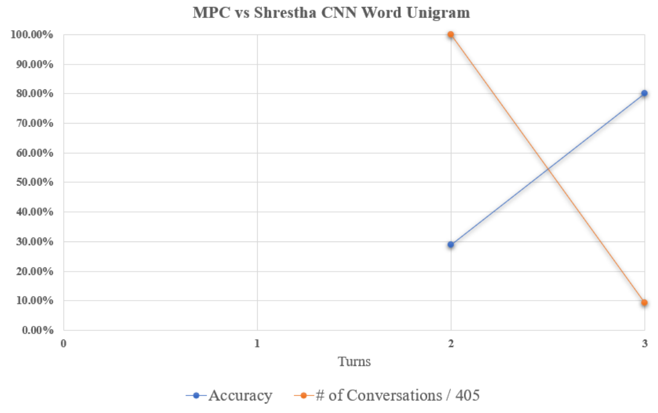

Fig. 10. MPC vs Shrestha’s word unigram (accuracy vs conversation)

VI. Discussion

In our experiments we used three different types of datasets to test the ability of the systems to identify the participants of the text. The first dataset was the C50 dataset that became the C25 dataset, which is not a conversation between two people. C25 is a dataset consisting of a text sample that is a combination of two independently written text by two different authors with the first authors text being followed by the second author’s text. When given this dataset, most of the systems seem to perform well. For the second dataset, we used an imbalanced scripted conversation dataset, Friends, to test how well they perform on conversations and the performance is poor. Finally, we used an imbalanced dataset consisting of actual conversations, MPC, to examine how well these systems can predict the participants in a conversation and the systems were mainly unsuccessful. In Experiments 3 and 4, our hypothesis is that when you increase the number of turns, thus increasing the sample size, the accuracy will increase. In this case, it also decreases the number of participants to choose from. This is also the case for MPC where the amount of turns was 2 and 3. At 3 turns, the accuracy increases for each of the top 5 algorithms along with the number of participants to choose from. The fluctuation seen in Friends dataset may be due to the variation in the author pairs and the reduction in the sample sizes. This could also be because Friends is a TV show that is written by at least 50 different screenwriters. In some cases, certain episodes even have more than one writer. Ideally, one would want each character to have their own personality which could be portrayed in a variety of ways by the screenwriters. Although the screenwriters are writing scripts for the same characters, each screenwriter’s writing style is different, and they could be a potential reason for the difficulty in identifying the conversation participants.

VII. Conclusion and Future Work

In this paper, two classes of methods were explored in an effort to identify the participants of a conversation. The first class explored was Speech Act Profiling (Single-Tagged and Multi-Tagged), the second class of methods explored was state-of-the-art authorship attribution systems. On the C50 and C25 datasets, most of the methods performed well, while all of the methods performed poorly on the Friends and MPC datasets. This may be due to the differences in the way we write, speak, and chat (write in chatrooms). The best performing algorithms were Teahan2003, Shrestha2017$_{\text{char-bigram,}}$ and Shrestha2017$_{\text{word-unigram.}}$

Our results also show that as the number of turns increases, so does the accuracy of identifying conversation participants. This comes at a cost of a reduced number of conversations within a dataset.

Our future work will be devoted towards the study of the effect that a balanced dataset has on the performance of the methods presented in this paper. Additionally, we will study how increasing the number of turns introduces imbalances within a dataset as the number of conversations is reduced.

References

[1] D. P. Twitchell and J. F. Nunamaker, "Speech act profiling: a probabilistic method for analyzing persistent conversations and their participants," *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the*, Big Island, HI, 2004, pp. 10 pp.-. doi: 10.1109/HICSS.2004.1265283

[2] D. P. Twitchell, M. Adkins, J. F. Nunamaker, and J. K. Burgoon, “Using Speech Act Theory to Model Conversations for Automated Classification Retrieval,” *Proceedings of the 9th International Working Conference on the Language-Action Perspective on Communication Modelling*, 2004.

[3] D. P. Twitchell, J. F. Nunamaker, and J. K. Burgoon, “Using Speech Act Profiling for Deception Detection BT - Intelligence and Security Informatics,” H. Chen, R. Moore, D. D. Zeng, and J. Leavitt, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 403–410

[4] D. Jurafsky, E. Shriberg, and D. Biasca, “Switchboard SWBD-DAMSL shallow-discourse-function annotation coders manual,” University of Colorado, Institute of Cognitive Science, Tech. Rep. Draft 13, 1997.

[5] V. Keselj, F. Peng, N. Cercone, and C. Thomas, “N-gram-based Author Profiles for Authorship Attribution,” in Proceedings of the Conference Pacific Association for Computational Linguistics, 2003, pp. 255–264.

[6] D. Benedetto, E. Caglioti, and V. Loreto, “Language trees and zipping,” Phys. Rev. Lett., vol. 88, p. 048702, Jan 2002. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.88.048702

[7] E. Stamatatos, “Authorship attribution based on feature set subspacing ensembles,” International Journal on Artificial Intelligence Tools, vol. 15, pp. 823–838, 2006.

[8] M. Koppel, J. Schler, and S. Argamon, “Authorship attribution in the wild,” Lang. Resour. Eval., vol. 45, no. 1, pp. 83–94, Mar. 2011. [Online]. Available: http://dx.doi.org/10.1007/s10579-009-9111-2

[9] W. J. Teahan and D. J. Harper, “Using Compression-Based Language Models for Text Categorization BT - Language Modeling for Information Retrieval,” W. B. Croft and J. Lafferty, Eds. Dordrecht: Springer Netherlands, 2003, pp. 141–165. [Online]. Available: https://doi.org/10.1007/978-94-017-0171-6_7

[10] E. Dauber, R. Overdorf, and R. Greenstadt, “Stylometric authorship attribution of collaborative documents,” in Cyber Security Cryptography and Machine Learning, S. Dolev and S. Lodha, Eds. Cham: Springer International Publishing, 2017, pp. 115–135.

[11] A. W. E. McDonald, S. Afroz, A. Caliskan, A. Stolerman, and R. Greenstadt, “Use fewer instances of the letter ”i”: Toward writing style anonymization,” in Proceedings of the 12th International Conference on Privacy Enhancing Technologies, ser. PETS’12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 299–318. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-31680-7_16.

[12] S. L. Condon and C. G. Cech, “Profiling turns in interaction: discourse structure and function,” in Proceedings of the 34th Annual Hawaii International Conference on System Sciences, Jan 2001, pp. 10 pp.–.

[13] H. Sacks, E. A. Schegloff, and G. Jefferson, “A simplest systematics for the organization of turn-taking for conversation,” *Language,* vol. 50, no. 4, pp. 696–735, 1974. [Online]. Available: https://muse.jhu.edu/content/crossref/journals/language/v050/50.4.sacks.html

[14] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, “Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit,” Nature, vol. 405, no. 6789, pp. 947–951, 2000. [Online]. Available: https://doi.org/10.1038/35016072

[15] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, “What is the best multi-stage architecture for object recognition?” in 2009 IEEE 12th International Conference on Computer Vision, Sep. 2009, pp. 2146–2153

[16] V. Nair and G. E. Hinton, “Rectified linear units improve re-stricted boltzmann machines,” in Proceedings of the 27th International Conference on International Conference on Machine Learning, ser. ICML'10. USA: Omnipress, 2010, pp. 807-814. [Online]. Available: http://dl.acm.org/citation.cfm?id=3104322.3104425

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, ser. NIPS'12. USA: Curran Associates Inc., 2012, pp. 1097-1105.[Online].Available:http://dl.acm.org/citation.cfm?id=2999134.2999257

[18] 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. DOI: https://doi.org/10.1145/3132039.

[19] M. G, “The One With the Un-cut Friends Scripts,” The One With the Un-cut Friends Scripts, 11-Mar-2004. [Online]. Available: http://uncutfriendsepisodes.tripod.com/. [Accessed: 11-Jul-2019].

[20] E. Shriberg, R. Bates, P. Taylor, A. Stolcke, D. Jurafsky, K. Ries, N. Coccaro, R. Martin, M. Meteer, and C. Van Ess-Dykema, “Can prosody aid the automatic classification of dialog acts in conversational speech?” Language and Speech, vol. 41, no. 3–4, pp. 439–487, 1998.

[21] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, M. Meteer, and C. Van Ess-Dykema, “Dialogue act modeling for automatic tagging and recognition of conversational speech,” Computational Linguistics, vol. 26, no. 3, pp. 339–371, 2000.

[22] J. Ziv and A. Lempel. 2006. A universal algorithm for sequential data compression. IEEE Trans. Inf. Theor. 23, 3 (September 2006), 337-343. DOI=http://dx.doi.org/10.1109/TIT.1977.1055714

[23] Y. Sari, A. Vlachos, and M. Stevenson, “Continuous n-gram representations for authorship attribution,” in Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp. 267–273. [Online]. Available: https://www.aclweb.org/anthology/E17-2043

[24] P. Shrestha, S. Sierra, F. González, M. Montes, P. Rosso, and T. Solorio, “Convolutional neural networks for authorship attribution of short texts,” in Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp.669–674.[Online].Available: https://www.aclweb.org/anthology/E17-2106

[25] S. Shaikh, T. Strzalkowski, G. A. Broadwell, J. Stromer-Galley, S. M. Taylor, and N. Webb, “Mpc: A multi-party chat corpus for modeling social phenomena in discourse.” in LREC, 2010.

[26] J. Houvardas and E. Stamatatos, “N-gram feature selection for authorship identification,” in Proceedings of the 12th International Conference on Artificial Intelligence: Methodology, Systems, and Applications, ser. AIMSA’06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 77–86. [Online]. Available: http://dx.doi.org/10.1007/11861461_10