Stylometry Based Authorship Identification

Mentor, Mrs. Sujata Khedkar
Associate Professor
Computer Engineering, VESIT, Chembur
sujata.khedkar@ves.ac.in

Shashank Agnihotri
B.E. Computer Engineering, VESIT, Chembur
shashank.agnihotri@ves.ac.in

Anshul Agarwal
B.E. Computer Engineering, VESIT, Chembur
anshul.agarwal@ves.ac.in

Mahak Pancholi
B.E. Computer Engineering, VESIT, Chembur
mahak.pancholi@ves.ac.in

Pooja Hande
B.E. Computer Engineering, VESIT, Chembur
pooja.hande@ves.ac.in

***Abstract*—"Every person is unique", we have been hearing this since ages. Every person has a unique identity, a unique fingerprint, a unique retina and a lot more. These features play a vital role in identification of individuals for security purposes. Unfortunately, when it comes to security of written pieces or words from an individual, these primary unique identities are futile. One cannot identify a writer from a written piece of text on the basis of retina or fingerprint scans, sometimes even the signature can be forged, in such situations for security purposes and intellectual property rights it becomes very important to identify the true author. Stylometry plays an important role in this. Every author has a unique style of writing, measure of this style of writing is called Stylometry. This paper proposes to identify authors from text based on their style of writing. First a data set consisting of articles, short stories and emails will be used to train the system for multiple authors, then a random text would be given to the system to identify the author correctly, if the author predicted by the system is similar to the author claimed then the information is authentic otherwise the author claiming to be the writer is a fraud. For stylometry, over the ages, many features have been focused on, but this paper proposes new features to be used for this purpose. While writing, there are many unconscious styles that are incorporated by the author, these features have been unnoticed till date, but can play a vital role in accurate and fast identification of authors. These features include: 'intellectual property right', 'chapter length', 'the importance of a word with respect to the other words in a document' and frequency of particular words per thousand words. The algorithms used to train the system can be Decision tree, Naive Bayesian or Multilayer Perceptron.**

Keywords—feature extraction, data set, Decision tree, artificial intelligence, machine learning, supervised learning, word2vec, sentence2vec, doc2vec.

I. Introduction

Various attempts have been made to identify author using stylometry. Most of the attempts made use of similar feature extractions but different data sets and algorithms. Every system had a drawback that couldn't be overlooked. Jose Hurtado, Napat Taweewitchakreeya, and Xingquan Zhu in their paper[1] used multilayer perceptron, random forest, SVM and k-nearest neighbour for training the data. Here the MLP learner, combined with the six categories of stylometric features, provides better performance over other classifiers and baseline approaches however Random forest and k-nearest neighbours give low accuracy and only few authors can be identified accurately. While in [2] Kohonen Self Organising Maps and backpropagation is used which is suitable to capture an intangible concept like style and in this fewer input variables are required as compared to the traditional statistics but this can be implemented only for small number of authors. [3] seems to cover all the drawbacks of [1] and [2] and other related works. [3] uses LDA and Naive Bayes for classification which enables it to do semantic analysis of corpus however it brings in a new drawback with it: to classify a new unknown document it would be necessary to reprocess all documents including new ones, this is an onerous and time consuming task. Thus this paper proposes a new methodology that encompasses almost all the benefits of [1], [2], [3] and [4] as it overcomes their

drawbacks. Stemming and Principal Component Analysis will provide a sharp edge in cutting down the processing time while increasing the efficiency of the new proposed system. Moreover focus on a new set of features will provide better accuracy and including a 'Pre-Processing stage' in the system will tremendously decrease the payload on the system when adding new data sets to the already trained system.

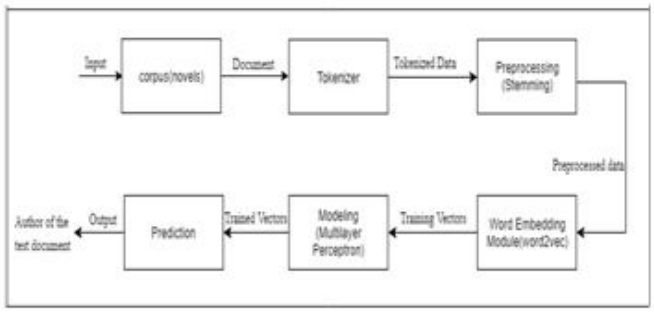

Fig 1. Block Diagram of final developed system

II. Preprocessing

A. Stemming

Stemming refers to a crude heuristic process which is commonly used to chop off the end of the words so as to achieve the desired goal easily and more correctly. It focuses on removing the derivational affixes as well.

Porter's Algorithm as mentioned in [6] can be used. 5 phases of word reductions are applied sequentially in Porter's algorithm. Each phase consists of various conventions to select the rules which are suitable.The example of the same can be that a rule can be selected from a particular rule group and hence applying it to the suffix with the largest length.

B. Data Cleansing

Data cleansing which is also known as data cleaning is the process in which we detect and correct the corrupt and records which are inaccurate from a record set, database or some table and then identifies inaccurate, incomplete, irrelevant or incorrect parts of the data and then modifying, deleting or replacing the dirty data. Data cleansing may be performed interactively with data wrangling tools, or as batch processing through scripting.

C. Principal Component Analysis

Principal Component Analysis refers to analysis of data which will be responsible to identify the patterns and then finding the patterns to reduce the dimensions of the dataset drastically, taking into consideration the minimal loss of the information. One of the way for performing Principal Component Analysis is by choosing a subset of Principal Components and Variables as mentioned in section 6 of [7].

III. Feature Extraction

A. Adopted Methods

1. *number of commas per thousand tokens:* Commas play a crucial role, which denote the ongoing flow of ideas within a sentence.

2. *number of ands per thousand tokens:* Ands are the markers used to represent coordination. It is frequently used in spoken production.

3. *number of buts per thousand tokens:* Buts are the markers of coordination, used to represent the contrastive linking.

4. *vocabulary:* Every authors selected vocabulary was chosen.

5. *number of colons per thousand tokens:* Colons indicate the reluctance of an author to stop a sentence where(s) he could.

6. *Frequency of words from bag-of-words:* The frequency of every word used is measured and words and their occurrence counts are mapped into categories of 'high frequency', 'mid frequency' and 'low frequency' for further processing.

7. *part-of-speech tagging (PoS tagging):* Penn Treebank PoS tagging denotes annotations. (ex. CC for coordinating conjunction, SVM for symbol)

8. *Word2vec:* The Authorship Attribution (AA) task consists in identifying the author of a given

text among a list of candidates authors. In this approach, the problem is treated as a supervised classification task, when a classifier is built using a training set and the task consists in classifying correctly the samples from a testing set. Word embeddings after cleaning the training data, we use the Word2vec method to obtain the vectors for each document. The Word2vec module offers two possible approaches to build the model, the Distributed Model (DM), which tries to predict the context of a given element and the Distributed Bag of Words (DBOW), which tries to predict the word given the context.

9. *Tf-idf:* In information retrieval, tf–idf or TFIDF, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in searches of information retrieval, text mining, and user modeling. The tf-idf value increases proportionally to the number of times a word appears in the document and is offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general. Nowadays, tf-idf is one of the most popular term-weighting schemes; 83% of text-based recommender systems in the domain of digital libraries use tf-idf.

IV. Methodologies adopted

A. *Method 1*

Algorithm:

1. Perform Stemming on the dataset.
2. Calculate the frequency of only lexical features* of the documents.
3. Divide the frequencies into 3 categories, low frequency, mid frequency (count 500 to 1000) and high frequencies.
4. Split the low, mid and high frequency features table into Training and Testing Data set.
5. Train the system on using Decision Tree Classifier, SVM and Neural Network.
6. Predict the authors for unseen features.

This approach gives us an accuracy of 37% to 90% but the system is not dynamic, and only lexical features are considered here which is not ideal.

B. *Method 2*

Algorithm:

1. Perform Stemming on the dataset.
2. Calculate the frequencies of lexical features of documents and bag-of-words.
3. Like Module 1 divide the frequencies into 3 categories, low frequency, mid frequency and high frequencies.
4. Perform Principal Component Analysis on mid frequency document.
5. Split the low, mid and high frequency features table into Training and Testing Data set.
6. Train the system on using Decision Tree Classifier, SVM and Neural Network.Predict the authors for unseen features.

This module gave us a low accuracy as compared to model 1 as PCA wiped out essential features being used in prediction. This approach gives us an accuracy of 30% to 82%. Moreover, here too only lexical features were being considered.

C. *Method 3*

Algorithm:

1. Perform stemming on the entire dataset
2. Calculate the “Term Frequency–Inverse Document Frequency” i.e. tf-idf score for the stemmed dataset.
3. Perform Principal Component Analysis (PCA) on the result of step 2.
4. Split the result table of step 3 into Training and Testing dataset.
5. Train the system using Decision Tree Classifier, SVM and MLP.
6. Predict the authors for unseen feature vectors.

This module gave us a very low accuracy of 10% to 32% because tf-idf score is not a very suitable approach for our dataset, which are large documents from many different authors which the number of documents per author varying a lot. Moreover, the system was static,

that is to test or train the system on a new file or author, the entire system had to be run again.

D. *Method 4*

Algorithm:

1. Perform Stemming on the dataset.
2. Calculate the frequencies of lexical features of the documents and bag-of-words.
3. Like Module 1 divide the frequencies into 3 categories, low frequency, mid frequency and high frequencies.
4. Perform Principal Component Analysis on mid frequency document.
5. Split the low, mid and high frequency features table into Training and Testing Data set.
6. Train the system using Decision Tree Classifier, SVM and Neural Network.
7. Perform k-fold cross validation on the dataset(k=10).
8. Predict the authors for unseen features.
9. Calculate the accuracy, the mean of the accuracy and standard deviation of the accuracy.

This module gave us a accuracy in the range of 31% to 85%. There was a bit drop in accuracy due to PCA.

E. *Method 5*

Algorithm:

1. Perform Stemming on the dataset.
2. Calculate the frequencies of lexical features of the documents and bag-of-words.
3. Like Module 1 divide the frequencies into 3 categories, low frequency, mid frequency and high frequencies.
4. Split the low, mid and high frequency features table into Training and Testing Data set.
5. Train the system using Decision Tree Classifier, SVM and Neural Network.
6. Perform k-fold cross validation on the dataset.
7. Predict the authors for unseen features.
8. Calculate the accuracy, the mean of the accuracy and standard deviation of the accuracy.

The only difference between Module 5 and Module 4 is performing PCA, however this small change had a huge impact of almost 7% - 8% on the accuracy of the system, Module 5 gave us an accuracy of around 41% to 92%.

F. *Method 6*

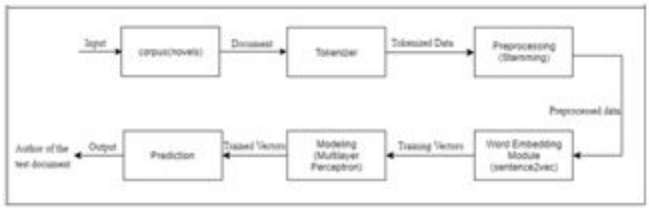

Fig 2: Model 6

Algorithm:

1. Tokenize dataset and perform stemming.
2. Perform Data cleansing and cleaning.
3. Perform sentence2vector operations on the cleaned dataset.
4. Obtain vectors for each sentence in a document.
5. Train model using these vectors and machine learning algorithms such as neural networks and SVM.
6. Test the trained model.

Accuracy by this model on a 5 author dataset trained using MLP is 68% and on a 15 author dataset is 49%.

G. *Method 7*

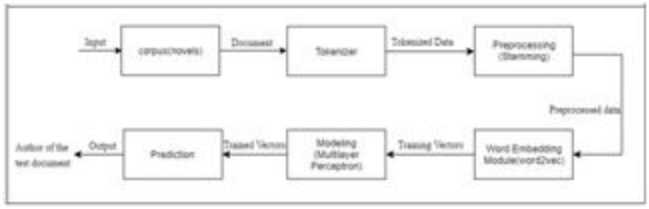

Fig 3: Model 7

Algorithm:

1. Tokenize dataset and perform stemming.
2. Obtain word2vec embeddings for each document.
3. Train model using these vectors and machine learning algorithms such as neural networks and SVM.
4. Test the trained model.

Accuracy by this model on a 15 author dataset trained using MLP is 83% . On 5 authors accuracy is 98.11%.

V. RESULTS

The system will calculate the chances of each author having the chances of writing the document, and the author which has the highest percentage would be identified by the system to be the true author. If the author claimed and the author identified by the system are same then the claim is validated, if not then the author has falsely claimed to be the author of that document.

Fig 4. Sample Output

This is how the result will look. There is a 78% probability that Martin was the author of the document given to the system for prediction, however there is a 32% probability that Edmund was the author too, while 5% probability of Edward being the author and some more smaller probabilities of some other known authors on which the system is trained to be the author of the document being tested.

As the probability of Martin being the true author of the system is the highest, based on the stylometry based tests, Martin is identified as the one true author of the document.

VI. COMPARISONS

Comparing the various built and tested models and their accuracies for the different machine learning algorithms the system was trained on. The primary machine learning algorithms used were SVM, Neural Network – Multilayer Perceptron (MLP) with 32 hidden layers and Decision Tree.

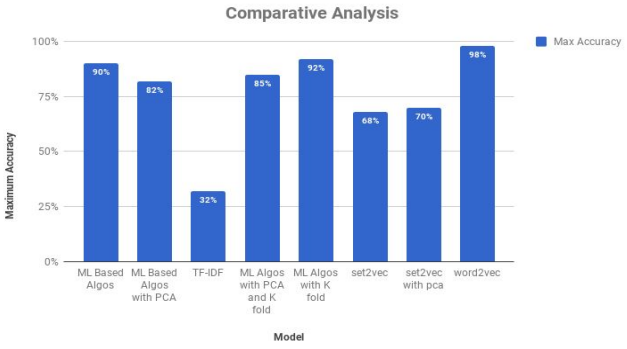

Fig 5. Sample Output

Parameters of SVM: (C=1.5, Degree=3, Kernel=poly)
Parameters of MLP: (hidden_layer_sizes (32,32) , alpha=0.001)
Type of Decision Tree: (CART). CART is an abbreviation for Classification & Regression Trees.

A. Model 1: Word Frequency based model without PCA

Algorithm	**Low Frequency**	**Mid Frequency**	**High Frequency**
SVM	57-62%	65-70%	**85-90**%
MLP	60-63%	60-65%	60-70%
Decision Tree	37-40%	37-42%	43-47%

Table 1: Comparisons of Model 1

B. Model 2: Word Frequency based model with PCA

Algorithm	**Low Frequency**	**Mid Frequency**	**High Frequency**
SVM	60-65%	64-67%	**77-82**%
MLP	53-56%	57-59%	54-58%
Decision Tree	30-32%	32-34%	34-37%

Table 2: Comparisons of Model 2

C. *Model 3: ML Based Model using TF-IDF*

Algorithm	**Accuracy**
SVM	22-32%
MLP	8-10%
Decision Tree	3-10%

Table 3: Comparisons of Model 3

D. *Model 4: Word Frequency based model with PCA and k-fold cross validation*

Algorithm	**Low Frequency**	**Mid Frequency**	**High Frequency**
SVM	60-66%	65-68%	**84-85**%
MLP	56-58%	70-73%	68-70%
Decision Tree	31-34%	35-37%	40-43%

Table 4: Comparisons of Model 4

E. *Model 5: Word Frequency based model without PCA and k-fold cross validation.*

Algorithm	**Low Frequency**	**Mid Frequency**	**High Frequency**
SVM	60-62%	75-77%	**90-92%**
MLP	66-68%	75-77%	67-75%
Decision Tree	41-45%	44-55%	47-52%

Table 5: Comparisons of Model 5

F. *Model 6: sentence2vector*

Algorithm	**Accuracy**
SVM	33%(5 authors) , 15%(15 authors)
MLP	**68%**(5 authors) , 49%(15 authors)
Decision Tree	41%(5 authors) , 22%(15 authors)

Table 6: Comparisons of Model 6

G. *Model 7: word2vector*

Algorithm	**Accuracy**
SVM	65%(5 authors),
MLP	**98.11%** (5 authors), 84%(15 authors)
Decision Tree	97.45%(5 authors), 45%(15 authors)

Table 7: Comparisons of Model 7

On comparing the results obtained by testing the above models it becomes very important to identify a single algorithm or model to adapt for the final Stylometry based Authorship Identification System being built.

Thus the maximum accuracies obtained from each developed model were collected and represented in graphical forms for better and clear understanding of the ideal model to be finally adopted. A close study of the accuracies revealed that the 'word2vec' model provided the highest accuracy to the system under development and thus after passing further tests developed using specific test cases, 'word2vec' was finally chosen.

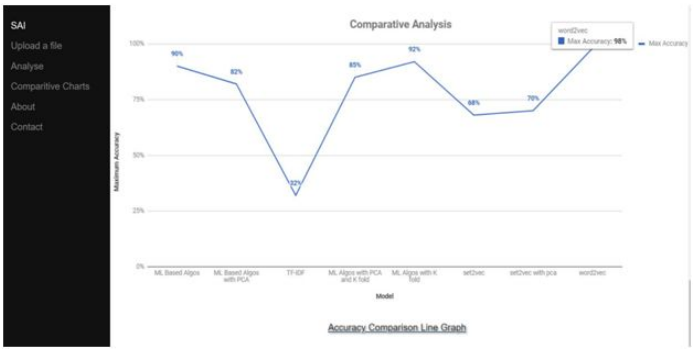

Fig.6 Accuracy comparison line graph

VII. CONSTRAINTS

With increasing communication and interaction between people, there are times that a single piece of document has not been written by a single author but has multiple authors, in such a case, the developed system fails to identify the multiple authors of the document and identifies the single largest probabilistic author as the true author of the document.

VIII. FUTURE SCOPE

Stylometry will play an important role in identification of potential social media hazards and in cracking cyber crime cases. Being able to incorporate short messages like tweets, Facebook posts or WhatsApp messages to train data and identify the author would be helpful and play an instrumental role in this field. This use of stylometry is also something that has to be left for future development as currently for accurately identifying authors there is substantial amount to written text that is needed to train the system.

Moreover, surpassing the constraint in the current system, of identifying documents written by not one, but multiple authors is another important aspect that can be added to the future scope of this developed system.

IX. CONCLUSION

Thus this paper addresses an old but unsolved problem of accurate and reliable author identification using stylometry. If successful, stylometry would a vital role in cybercrime forensics and would help the world solve ages of mysteries regarding ownership of various writing pieces by authors. It would be used to identify anonymous works and saying by comparing it the the style of authors and famous personalities of those days.

For this very purpose, the paper proposed new features to be extracted from the document which hopefully would assist in increasing accuracy and reducing a few redundant dimensions. The paper even proposes a new algorithm that could be identified for faster computation and better accuracy. This new proposed algorithm is basically a combination of two well known and used algorithm. This paper certainly proposes methods that would overcome the known drawbacks of previous works in this field.

REFERENCES

[1] Hurtado, Jose, Napat Taweewitchakreeya, and Xingquan Zhu. "Who wrote this paper? learning for authorship de-identification using stylometric features." Information reuse and integration (IRI), 2014 IEEE 15th international conference on. IEEE, 2014.

[2] Ramyaa, Congzhou He, and Khaled Rasheed. "Using machine learning techniques for stylometry." Proceedings of International Conference on Machine Learning. 2004.

[3] Hernández-Castañeda, Ángel, and Hiram Calvo. "Author Verification Using a Semantic Space Model." Computación y Sistemas 21.2 (2017).

[4] Brocardo, Marcelo Luiz, et al. "Authorship verification for short messages using stylometry." Computer, Information and Telecommunication Systems (CITS), 2013 International Conference on. IEEE, 2013.

[5] Crawford, Michael, et al. "Survey of review spam detection using machine learning techniques." Journal of Big Data 2.1 (2015): 23.

[6] Willett, Peter. "The Porter stemming algorithm: then and now." Program 40.3 (2006): 219-223.

[7] Jolliffe, Ian T. "Principal Component Analysis and Factor Analysis." Principal component analysis. Springer New York, 1986. 115.

[8] Krause, Markus, "Stylometry-based Fraud and Plagiarism Detection for Learning at Scale", 2015 5th KSS Workshop, Karlsruhe, Germany

[9] P. Das, R. Tasmim and S. Ismail, "An experimental study of stylometry in Bangla literature," 2015 2nd International Conference on Electrical Information and Communication Technologies (EICT), Khulna, 2015, pp. 575-580.

[10] Ramnial H., Panchoo S., Pudaruth S, "Authorship Attribution Using Stylometry and Machine Learning Techniques", 2016 Intelligent Systems Technologies and Applications. Advances in Intelligent Systems and Computing, vol 384. Springer, Cham

[11] Maciej Eder, Jan Rybicki and Mike Kestemont , "Stylometry with R: A Package for Computational Text Analysis",The R Journal Vol. 8/1, Aug. 2016

[12] Lakshmi, Pushpendra Kumar Pateriya, "A Study on Author Identification through Stylometry ",Lakshmi et al , International Journal of Computer Science & Communication Networks,Vol 2(6), pp. 653-657

[13] Ganapathi N V Raju, Ch. Sadhvi, P Tejaswini and Y Mounica, "Style based Authorship Attribution on English Editorial Documents", International Journal of Computer Applications 159(4), pp. 5-8, February 2017.

[14] A. Rocha et al., "Authorship Attribution for Social Media Forensics," in IEEE Transactions on Information Forensics and Security, vol. 12, no. 1, pp. 5-33, Jan. 2017.

[15] Helena G´omez-Adorno, Posadas-Dur´an, Juan-Pablo, Grigori Sidorov, David Pinto," Document Embeddings Learned on Various Types of n-grams for Cross-Topic Authorship Attribution" , in Computing 2018, pp. 1-16.

[16] Đlker Nadi Bozkurt, Özgür Bağlıoğlu, Erkan Uyar, "Authorship Attribution Performance of various features and classification methods", in Proceedings Bozkurt Authorship AP, 2017.