IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Study Of Various Problem-Solving Approaches In Authorship Attribution Using Stylometry

[1]Urmila Mahor, [2]Aarti Kumar
[1]Ph.D. Scholar, [2]Professor
[1]Department of Computer Science and Engineering,
[1] Rabindranath Tagore University, Bhopal, Madhya Pradesh, India

Abstract: As we know that each author has its own writing habits and certain inherent writing characteristics, that he or she uses in his or document writing unconsciously, we can study those hidden characteristics to identify their style of writing, this is very helpful in case of authorship detection, while two or more persons claim the ownership for the same content or document. Now a days, due to digitalization, someone can copy and paste your document and publish it with own name, which is a matter of copyright, but without sufficient proof, it is difficult to prove true legitimate ownership, in this paper we are discussing various characteristics and approaches and their importance that were used and discussed by many researchers in their work.

Index Terms **- : Features, classification, authorship attribution, stylometry.**

I. INTRODUCTION

Stylometry has the power to distinguish the authorship of a document in a text. Mosteller and Wallace used frequencies of function words because function word represents the grammatical relationship among the words in a sentence and define syntactic relationship in the sentence. Since function words are topic-independent and can be considered an efficient and effective measure for the AA. Authorship attribution plays a vital role in the field of stylometry or the computational analysis of writing style [12,27,28]. Stylometric features generally involve the inherent characteristics of a document, which appear unconsciously in the author's writing style. These features may be quantifiable and salient and cannot be manipulated [14]. The accuracy of authorship attribution methods depends not merely on selected methods, but also depends on the training and test data, the size of data, number of features. In our study we found that many researchers have used lexical, syntactical, function words, vocabulary richness, grammatical sequences, prefix and suffixes, fingerprints, POS as stylometric features [2,4,11,23,27,30].

II. FEATURES

In literary authorship, stylometric features are commonly used. Some examples of stylometric features are summarized in the list below.

- Letter frequencies
- Character N-gram frequencies (overlapping n-character frequencies)
- Function word usage (short structure-determining words: common adverbs, auxiliary verbs, conjunctions, determiners, numbers, prepositions, and pronouns)
- Vocabulary richness (number of different words used)
- Lexical richness (word frequency as a function of full text)
- Distribution of syllables per word
- Word frequencies
- Hapax legomena (words used once only)
- Hapax dislegomena (words used twice only)
- Word length distribution
- Word collocations (words frequently used together)
- Sentence length
- Preferred word positions
- Prepositional phrase structure

- Distribution parts of speech
- Phrasal composition grammar
- FOG Index
- SMOG Index
- Readability Index
- Vocabulary Density

III. CLASSIFICATION APPROACHES

Machine learning algorithms learn the characteristics of training data samples. This information is often used to create a model. In essence, this is a classification model; each combination of different feature values for the characteristics is labeled with a predefined class. The model is then used to generalize over unseen data. The model uses the characteristics of the unseen data to predict the class label for this unseen data sample. The unseen data sample receives the class label predicted by the model. Different types of machine learning algorithms achieve this differently.

Different machine learning algorithms provide different classification results. For author identification, different methods are used, like support vector machines and neural networks. There is no consensus on which is the best classification method to be used for authorship identification; however, support vector machines are widely used. In this paper, we will discuss the Decision Tree, Nearest Neighbor, Neural Network, and Support Vector Machine algorithms that may be suitable for the classification problem of identifying an author, given a list of possible authors. The results are compared to determine which algorithm is most suitable for author identification.

3.1 Naive Bayes classifier

The Naive Bayes Classifier technique is based on the so-called Bayesian theorem and is particularly suited when the dimensionality of the inputs is high. Despite its simplicity, Naive Bayes can often outperform more sophisticated classification methods. A naive Bayes classifier builds a probabilistic model of each authorship class based on the training data of that class. Then it calculates and multiplies the probabilities of all features to give the probability of the test text. The highest probability among all authors is most likely an author of that anonymous or test text.

Naive Bayes classifiers have been used for authorship attribution in many languages, including English [13,32]. The disadvantage of the Naive Bayes is that when the test data contains features in which the model has not been seen in training data. So some probabilities yield zero-result since none of the training data falls in the range. These zero counts have a zero probability, leaving the Naive Bayes classifier unable to predict a class.

Table 1: Describes the use of Naïve Bayes Classifier with various characteristics

Author's Name	Year	Features or Characteristics
Mosteller and Wallace	1964	Function words
Imene Bensalem et. Al	2014	Character n-gram
Jurgita kapociute-dzikiene et al	2017	Word or character n-gram
P. Jeevan Kumar et al	2017	Term weight measure, BOW
Palacharla Ravikumar et al	2020	Word n-gram, POS
Jagadeesh Patchala et al	2018	Syntactic features
Fatma Howedi et al	2014	Lexical, character n-gram word n-gram
A. Pian et al	2019	N-gram, parts-of-speech, function words, semantic Features
Clement, Sharp	2003	Character n-gram
Peng et.al.	2004	Character n-gram, word n-gram
Zhao, Zobel	2005	Function words,
Aylin caliskan et al	2012	Function words, character grams, part of speech tags, word length, words, sentence length, word grams, rare words
Al-Falahi Ahmed et al	2019	Character n-gram, function words

3.2 Nearest Neighbor classifier

The Nearest Neighbor rule achieves consistently high performance, without a priori assumptions about the distributions from which the training examples are drawn. It involves a training set of both positive and negative cases. The nearest neighbor algorithm does not create a model from the training data. So, this algorithm is called `lazy learner'. It retrieves the information from the test data when needed to classify an unseen sample. Each sample from the training data is represented as an n-dimensional data point. The `n' represents the number of features that are used to describe the data. When an unseen sample is presented to the algorithm it will retrieve the k-nearest neighbors of that sample calculated with a proximity measure. The `k' is the number of nearest neighbors that should be retrieved. The unseen data sample gets the same class label as its k neighbors. If

these neighbors have more than one class the unseen data receives the label that the majority of its neighbors have. If there is a tie between class labels, a random class label is given to the unseen sample.

The Nearest Neighbor algorithm does not create a model from the training data. This algorithm is a so-called `lazy learner', it retrieves the information from the test data when needed to classify an unseen sample. Each sample from the training data is represented as an n-dimensional data point. The `n' represents the number of features that are used to describe the data. When an unseen sample is presented to the algorithm it will retrieve the k-nearest neighbors of that sample calculated with a proximity measure.

The `k' is the number of nearest neighbors that should be retrieved. The unseen data sample gets the same class label as its k neighbors. If these neighbors have more than one class the unseen data receives the label that the majority of its neighbors have. If there is a tie between class labels, a random class label is given to the unseen sample.

A disadvantage of the nearest neighbor algorithm is that when there are a lot of features many examples are needed to perform the classification. For the domain of author identification, this will be a problem when many messages are used, which results in a lot of features.

Table 2: Describes the use of Nearest Neighbor Classifier with various characteristics

Author's Name	**Year**	**Features or Characteristics**
Mathews, Merriam	1993	Function words
Merriam, Mathews	1994	Function words
Kjell	1994	Character n-gram
Kjell et al.	1995	Character n-gram
Baayen et al.	1996	Syntactic features
Tweedie et. al.	1996	Function words
Hoorn et.al.	1999	Character n-gram
Waugh et.al.	2000	Words n-gram
Zheng et.al.	2006	Characters, function words, syntax, vocabulary richness
Oren Halvani et al	2018	Character n-gram

3.3 Neural Networks classifier

A neural network is made up of nodes with directed weighted links between them. The network has an input layer representing the input features, an output layer to give the output of the model, and possibly several hidden layers. The weighted sum of the input of a node is used as an input for an activation function, which determines the output of that node. The activation function makes it possible to produce an output that is a nonlinear function of the inputs. During the learning phase, the weights of the network are adjusted until an error rate is minimized. A widely used method to minimize this error is gradient descent. For training the hidden units a commonly used method is back-propagation, [17] used a neural network with character bigrams as features to identify the authors of articles in the Federalist Papers. The disadvantage of a neural network is a lot of parameters have to be set, the number of input nodes which depends on the number and type of features, the number of output nodes which depends on the number of classes, the number of hidden layers, number of nodes in the hidden layers, the activation function, and the initial weights. Improperly setting these parameters may result in under-fitting so the network cannot fully describe the data or in over-fitting, so the network cannot generalize well to unseen data.

Table 3: Describes the use of Neural Network Classifier with various characteristics

Authors' Name	**Year**	**Features**
Martinendale, McKenzie	1995	Words n-gram
Baayen et al.	1996	Syntactic features
Waugh et.al.	2000	Words n-gram
Sebastian Ruder et al	2016	N-gram
Dainis Bomber et al	2016	N-gram
Aisha Khatun et al	2020	Character n-gram
HainingWang et al	2021	Stylometric features
K. A. Apoorva , S. Sangeetha	2021	Stylometry features
Aleks, Romanov et al	2021	Stylometry features

3.4 Support Vector Machines classifier

This technique is based on finding the maximal margin hyper-plane which separates the data into two sets. Finding this hyper-plane is based on structural risk minimization, a principle that tries to minimize the generalization error while minimizing the training error and avoiding a model that is too complex. The earlier discussed machine learning techniques only minimized the training error, but this does not necessarily mean that the generalization error is minimized. So theoretically this means that SVM can better generalize over unseen data. The standard authorship attribution in which we need to assign an anonymous document to one of a small closed set of candidates is well understood and has been summarized in several surveys [28]. A binary learning problem and SVM has often been found to perform well for binary authorship problems [2,31] .

This technique is based on finding the maximal margin hyper-plane which separates the data into two sets. Finding this hyper-plane is based on structural risk minimization, a principle that tries to minimize the generalization error while minimizing the training error and avoiding a model that is too complex. The earlier discussed machine learning techniques only minimized the training error, but this does not necessarily mean that the generalization error is minimized. So theoretically this means that SVM can better generalize over unseen data. And in contrast with decision trees and neural networks, SVM does not use a Greedy approach, therefore it can find the globally optimal solution.

An SVM tries to find the hyper-plane with the largest margin because this improves the generalization error. A small margin is prone to over-fitting. The hyper-plane is positioned so that the margin between the classes is as large as possible. Only the data points that are necessary to determine the largest margin are considered, these are called the support vectors. Note that other possible hyper-planes could separate this data, but for these hyper planes the margins are smaller. In cases where the data is not linearly separable a soft margin approach can be used. This approach makes a trade between the width of the margin and the number of training errors. There are also cases in which classes are separated by a nonlinear boundary. For these cases, the Kernel trick can be used. With a Kernel, trick data is mapped into a new space in which a linear hyperplane can be found.

Table 4: Describes the use of SVM Classifier with various characteristics

Authors' Name	Year	Features
Marcia Fissette	2010	Word unigrams, bi-grams
Navot Akiva	2012	Bow
Jan Rygl	2013	Sentence length, word length, vocabulary richness, punctuation marks
Moshe Koppel et.al.	2014	Min-max matrix, bow, character-tetra grams
Jan Rygl	2014	Capital letters, sentence length distribution, frequency of emoticons, morphological tags, morphological categories, frequency of stop words, syntactic analysis, typographic errors, n-gram syntactic analysis, punctuation analysis, vocabulary richness, word repetition analysis, frequency of word classes, freq. of word-class bigrams
Daniel Castro	2015	Layers phonetic, character, lexical, syntactic, , semantic
Michael Tschuggnall et al	2019	Distributed bag of words
Hans van Halteren	2019	N-gram, POS
HainingWang et al	2021	Stylometric features

3.5 LDA classifier

Latent Dirichlet Allocation (LDA) [5] to build models of authors from their texts. LDA is a generative probabilistic model that is traditionally used to find topics in textual data. The main idea behind LDA is that each document in a corpus is generated from a distribution of topics, and each word in the document is generated according to the per-topic word distribution. [5] Showed that using LDA for dimensionality reduction can improve performance for supervised text classification. We know only one case where LDA was used in authorship attribution reported preliminary results on using LDA topic distributions as feature vectors for SVMs, but they did not compare the results obtained with LDA-based SVMs to those obtained with SVMs trained on tokens directly.

Table 5: Describes the use of LDA Classifier with various characteristics

Authors' Name	Year	Features
Morton	1956	Sentence length
Stamatatos	2001	Syntactic features
Chaski	2005	Character and word n-gram, POS
Goksel	2012	Bow
Steven H.H. Ding et al	2016	Lexical, syntactic, and character level
David Kernot et al	2017	Richness, personal pronouns, referential activity power, sensory based adjectives , words
Al-Falahi Ahmed et al	2019	Character n-gram, function words

IV. CONCLUSION

In our study, we studied various research papers and found that researchers used classifiers according to their data types, we also noticed that Support Vector Machine is used mostly by researchers in their work. Every algorithm has its advantages and dis-advantages, Performance of a classification algorithm also depends on the feature vector size and usefulness of features. so we cannot say that one algorithm is sufficient and applicable to all problems. Because the performance of the classifier depends on data type and size.

REFERENCES

[1] Akiva, N., Koppel, M. 2012. Identifying Distinct Components of a Multi-Author Document. pp. 205–209.
[2] Abbasi, Chen, H. 2008. Writeprints: A Stylometric Approach to Identity-Level Identification and Similarity Detection in Cyberspace. ACM Transactions on Information Systems. pp. 1–29.
[3] Baayen, H., Van Halteren, H., and Tweedie, F. 1996. Outside the Cave of Shadows: Using Syntactic Annotation to Enhance Authorship Attribution, Literary and Linguistic Computing. pp. 121–31.
[4] Baayen, H., Van Halteren, H., Neijt, A., Tweedie, F. 2002. An Experiment in Authorship Attribution.
[5] Blei D.M, and Andrew Y.N. 2003. Latent Dirichlet Allocation. Journal of Machine Learning Research 3(4-5):993-1022 DOI:10.1162/jmlr.2003.3.4-5.993.
[6] Chaski, E. 2003. Who's at the Keyboard: Authorship Attribution in Digital Evidence Investigations.
[7]Ding, S.H.H. and Fung, B. C. M.2016. Learning Stylometric Representations for Authorship Analysis. Institute for Linguistic Evidence.
[8] Howedi, F., Mohd, M..2014. Text Classification for Authorship Attribution Using Naive Bayes Classifier with Limited Training Data. Computer Engineering and Intelligent Systems ISSN 2222-1719, [Paper] ISSN 2222-2863 , vol.5, no.4.
[9] Feiguin,O. and Hirst, G.2007.Authorship attribution for Small Texts: Literary and Forensic Experiments. Proceedings of the SIGIR 2007. International Workshop on Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection, PAN 2007, Amsterdam, Netherlands.
[10] Gamon, M.2004.Linguistic Correlates of Style: Authorship Classification with Deep Linguistic Analysis Features. Proceedings of the International Conference on Computational Linguistics. pp. 611–617.
[11] Halteren, H., Baayen, H. R., Tweedie, F.,Haverkort, M., and Neijt, A. 2005. New Machine Learning Methods Demonstrate the Existence of A Human Style. Journal of Quantitative Linguistics, 12(1): 65–77.
[12] Hoorn, J., Frank, S.,Kowalczyk, W, and Van Der Ham, F. 1999. Neural Network Identification of Poets Using Letter Sequences, Literary and Linguistic Computing, 14(3):311-338.
[14] Holmes. 1994. Authorship attribution, Computers and the Humanities, 28(2): 87–106.
[15] Holmes , G. and Nevill-Manning, C.G.1995. Feature Selection via the Discovery of Simple Classification Rules, In Proceedings of the Symposium on Intelligent Data Analysis. Baden- Baden, Germany. Holte, R. C. Very simple classification rules perform well on most commonly used datasets. Machine Learning, pp. 63–91.
[16] Rygl, J. Horak, A.2012. Similarity Ranking as Attribute for Machine Learning Approach to Authorship Identification. Proceedings of the Eighth International Conference on Language Resources and Evaluation. pp 726–729.
[17] Kjell, B. 1994. Discrimination of Authorship Using Visualization. Information Processing and Management. 30(1): pp.141-150.
[18] Koppel, M., Schler, J., and Argamon, S.2010. Authorship Attribution in the Wild. Language Resources and Evaluation. Advanced Access published January 12, 2010:10.1007/s10579-009-9111-2.
[19] Koppel, M. 2008. Computational Methods in Authorship Attribution, JASIST, 60 (1): 9-26.
[19] Koppel M. 2014. Winter Y. Determining if Two Documents are by the Same Author. J. Am. Soc. Inf. Sci. Technol. 65, pp. 178–187.
[20] Luyckx, K. and Daelemans W.2008. Authorship Attribution and Verification with Many Authors and Limited Data. In Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1, ser.COLING '08. Stroudsburg, PA, USA: Association for Computational Linguistics.pp. 513–520.
[21] Matthews,R. A. J. and Merriam,T. V. N. 1993. Neural Computation in Stylometry. An application to the works of Shakespeare and Marlowe, Literary and Linguistic Computing. vol. 8, no. 4, pp. 203–209.
[22] Mosteller and Wallace, D.1964. Inference and Disputed Authorship: The Federalist. Series in Behavioral Science: Quantitative Methods Edition. Reading, MA: Addison-Wesley.

[23]Peng, F., Shuurmans, D., and Wang, S. 2004 Augmenting Naive Bayes Classifiers with Statistical Language Models, Information Retrieval Journal. 7(1): 317-345.
[24] Merriam,T. V. N. and Matthews,R. A. J.1994. Neural Computation in Stylometry II: An application to the works of Shakespeare and Marlowe. Literary and Linguistic Computing, vol. 9, no. 1, pp. 1–6.
[25] Peng , Hengartner.2002. Quantitative Analysis of Literary Styles, The American Statistician 56(3) DOI:10.1198/000313002100.
[26] Stamatatos, E., Fakotakis, N., and Kokkinakis, G.2000. Text Genre Detection Using Common Word Frequencies, Proceedings of the 18th International Conference on Computational Linguistics, vol. 2.Saarbrucken, Germany. Association for Computational Linguistics, pp. 808–14.
[27] Stamatatos, E.2009.A Survey of Modern Authorship Attribution Methods, Journal of the American Society for Information Science and Technology 60, 3.
[28] Stamatatos, E. 2001. Computer-based Authorship Attribution without Lexical Measures, Computers and the Humanities 35: 193-214.
[29] Tweedie, F., Singh, S., and Holmes, D. 1996. Neural Network Applications in Stylometry: The Federalist Papers. Computers and the Humanities, 30(1):1-10.
[30] Zheng, R., Li, J.,Chen, H., and Huang, Z.2006. A Framework for Authorship Identification of Online Messages: Writing-Style Features and Classification Techniques. J. Am. Soc. Inf. Sci. Technol.
[30] Zhao,Y.and Zobel, J.2005. Effective and Scalable Authorship Attribution using Function Words. Proceedings of the 2nd Asia Information Retrieval Symposium. Jeju Island, Korea: Springer, pp. 174–90.