Author Detection using Stylometric Analysis

Parth Patwa - S20170010106
Tunuguntla Ooha -S20170010167
Moosa Mohamed - S20170010096

1. Introduction

Detection of the author of a written piece of document is an important problem in industry and academia. Many older documents have disputed claims of authorship. Some documents have no known author and the author is believed to be dead. In such scenarios, we need to use stylometric analysis to detect the author. We can look into the writing style, use of punctuations, vocabulary etc.

Sometimes, we have multiple documents written by multiple unknown authors. In this case, we need to segregate the documents into clusters. It is possible that we don't have any information about the potential authors. In this case, we need to segregate the documents only based on the text and no previous information. This needs to be solved using unsupervised learning.

In this project we do the following things :

1) Author detection from non technical documents
2) Author detection from technical documents
3) Unsupervised clustering of documents

2 Data

We use the following datasets:

2.1 Horror stories dataset

This dataset contains roughly 20,000 horror stories written by 3 authors. We use this data as non technical documents.

2.2 Federalist papers dataset

This dataset contains 85 political articles written in the 1700s. They contributed towards the US constitution. They were written by 3 republicans - Alexander Hamilton, James Madison, John Jay. Some of these papers have disputed authorship. We use this data as technical documents.

2.3 Excerpts from story and technical paper

For unsupervised clustering, we use excerpts from a children's story and a technical paper.

3 Method:

3.1 Non technical documents

To detect authors of non technical documents, we extract lexical features, syntactic features and character level features. We use a combination of these features as input to various (explainable and easy to understand) Machine Learning algorithms. We use supervised learning. The data is divided into 80-20 train test splits.

Features

i) Character features - Characters, character n grams
ii) Syntactic features - Parts of Speech
iii) Lexical Features - Tokens, word n grams, Tf-Idf score

TFIDF - it is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. This is done by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents.

Algorithms

We use the following Machine Learning Algorithms:

- **Logistic Regression (LR)** - It gives the probability of a document belonging to a particular author. It uses the Logistic Loss function.
- **Decision Tree (DT)** - It builds a tree by coming up with a lot of if-else questions.
- **Gradient Boosting (GDBT) -** It ensembles a lot of weak decision trees.
- **Support Vector Machine (SVM) -** It is also called a large margin classifier**.** It tries to find boundaries to separate documents.

3.2 Technical Documents

Features:

Mendenhall's Characteristic Curves of Composition - T. C. Mendenhall wrote that an author's stylistic signature could be found by counting how often he or she used words of different lengths. For example, if we counted word lengths in several 1,000-word or 5,000 word segments of any novel, and then plotted a graph of the word length distributions, the curves would look pretty much the same no matter what parts of the novel we had picked.

Kilgariff Chi-Squared Method-Chi-square is used to test whether a set of observations follow a certain probability distribution or pattern. we will simply use the statistic to measure the distance between the vocabularies employed in two sets of texts. The more similar the vocabularies, the likelier it is that the same author wrote the texts in both sets. This assumes that a person's vocabulary and word usage patterns are relatively constant. In the below formula C(i) represents the observed number of tokens for feature 'i', and E(i), the expected number for this feature.

$$\chi^2 = \sum_i \frac{\left(C_i - E_i\right)^2}{E_i}$$

John Burrows' Delta Method -It is a measure of the distance between a text whose authorship we want to ascertain and some other corpus. Unlike chi-squared, however, the Delta Method is designed to compare an anonymous text to many different authors' signatures at the same time. More precisely, Delta measures how the anonymous text and sets of texts written by an arbitrary number of known authors all diverge from the average of all of them put together.Below figures shows the z-score and delta of candidate c for feature 'i', where C(i) represents the observed frequency, the greek letter mu represents the mean of means, and the greek letter sigma, the standard deviation and Z(c,i) is the z-score for feature 'i' in candidate 'c', and Z(t,i) is the z-score for feature 'i' in the test case.

$$Z_i = \frac{C_i - \mu_i}{\sigma_i} \qquad \Delta_c = \sum_i \frac{\left| Z_{\mathrm{c(i)}} - Z_{\mathrm{t(i)}} \right|}{n}$$

The winning candidate is the author for whom the delta score between the author's subcorpus and the test case is the lowest.

3.3 Unsupervised Clustering of Documents

Here, we don’t have a dataset of documents with author names. We will build a model which won’t have the names of authors. It will only take documents as input and differentiate them on the basis of writing style and features.

Features

- **Lexical features -** Average Word Length, Average Sentence Length By Word, Average Sentence Length By Character, Special Character Count, Average Syllable per Word, Functional Words Count, Punctuation Count

- **Vocabulary Richness Features -** These features tell us about the diversity and richness of the vocabulary used in the text.
    - **Hapax Legomena** is a word that occurs only once within a context, either in the written record of an entire language, in the works of an author, or in a single text.
    - **Shannon Entropy** tells us the about the disorder in a system
    - **Simpson's Diversity** Index is a measure of diversity. In ecology, it is often used to quantify the biodiversity of a habitat.

- **Readability features -** Readability is the ease with which a reader can understand a written text. Readability is more than simply legibility—which is a measure of how easily a reader can distinguish individual letters or characters from each other.
    - **Flesch Reading Ease** tells us roughly what level of education someone will need to be able to easily read a piece of text. The Reading Ease formula generates a score between 1 and 100.
    - In linguistics, the **Gunning fog index** is a readability test for English writing. The index estimates the years of formal education a person needs to understand the text on the first reading.

Algorithms

- **PCA -** In order to visually see those clusters we convert our 20 dimensional vector into a 2D vector using Principal Component Analysis which extracts the essence from that 20D vector and converts it into a 2D vector. We then plot these vectors and color those chunks which are grouped together under a centroid by K-Means. This way the chunks with different styles are visualized further strengthening our results.
- **K-means clustering -** We use the K-Means algorithm to identify K different centroids in a text having different writing styles. Each centroid spans those chunks which have the same writing style. Hence the number of centroids correspond to the different number of writing styles that a document has.

- **K value and Elbow method -** It is necessary to know the value of K beforehand to run K-means effectively. We hence use the elbow method to find the optimal value of K. Compute the sum of squared error (SSE) for some values of k (for example 2, 4, 6, 8, etc.). The SSE is defined as the sum of the squared distance between each member of the cluster and its centroid. We then plot ‘k; against the SSE, we will see that the error decreases as k gets larger; this is because when the number of clusters increases, distortion gets smaller. The idea of the elbow method is to choose the k at which the SSE decreases abruptly. This produces an "elbow effect" in the graph, as can be seen in the figure under section 4.3

4) Results

4.1 Non technical Documents

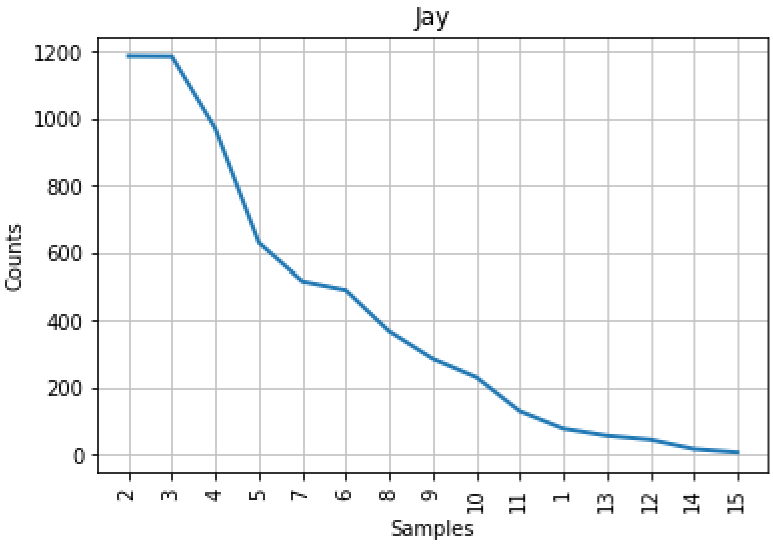
We report the results on the test set. Only results for best model per feature input are shown. For all results, please check the appendix.

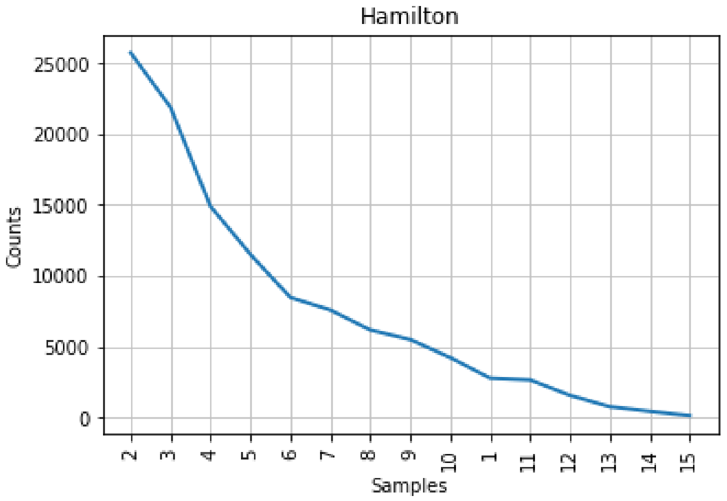
Features input	Best Algorithm	Accuracy	F1 score
Tokens	SVM	80.26	80.26
Tokens, tf-idf	SVM	83.57	83.58
Tokens , postagging	LR	40.23	38.36
Tokens, tf-idf, n-gram	**SVM**	**83.83**	**83.83**
Chars	GDBT	58.65	59.77
Chars, tf-idf	GDBT	58.40	59.13
Chars, tf-idf, n-gram	SVM	69.25	69.29

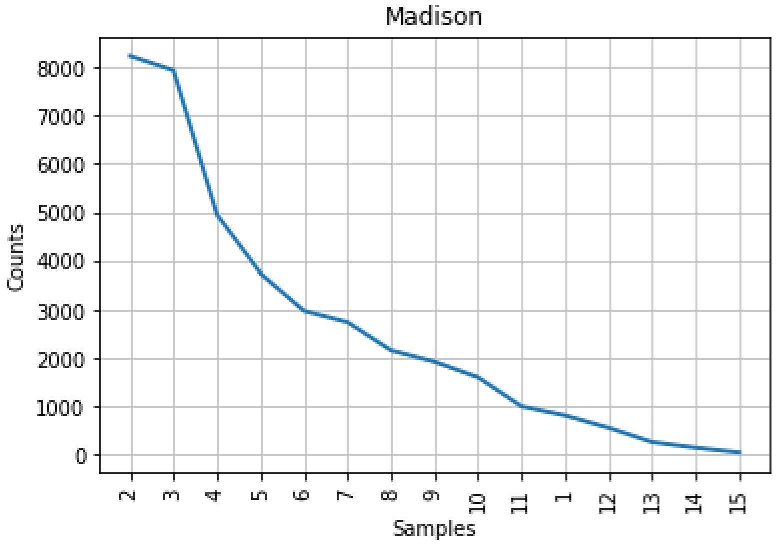
Caption: Test results of the best algorithm on various feature combinations.

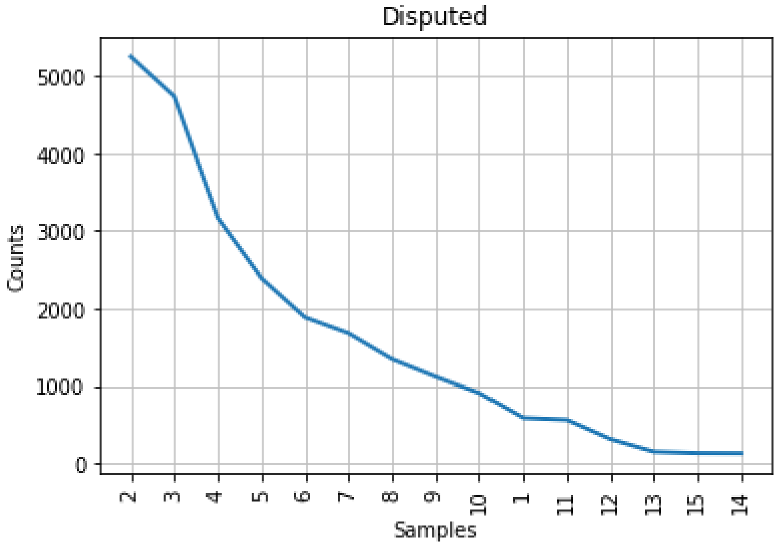
4.2) Technical Documents

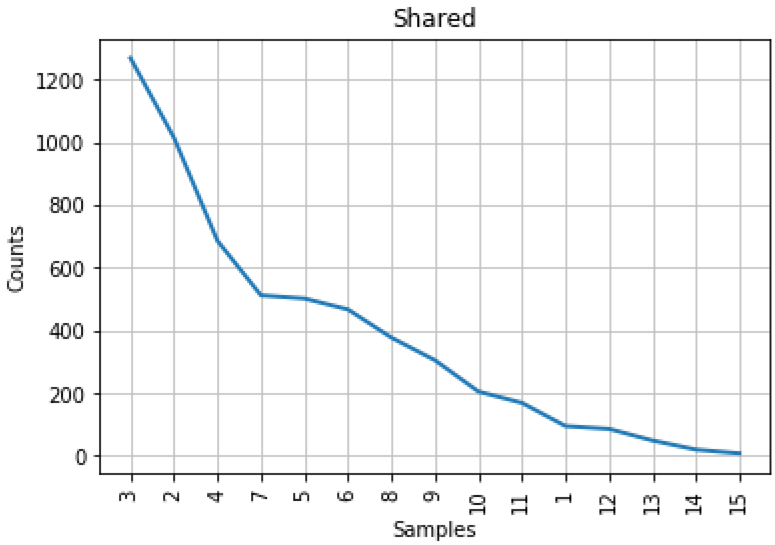
Plots for Mendenhall’s Characteristic Curves




Hamilton


Madison


Disputed



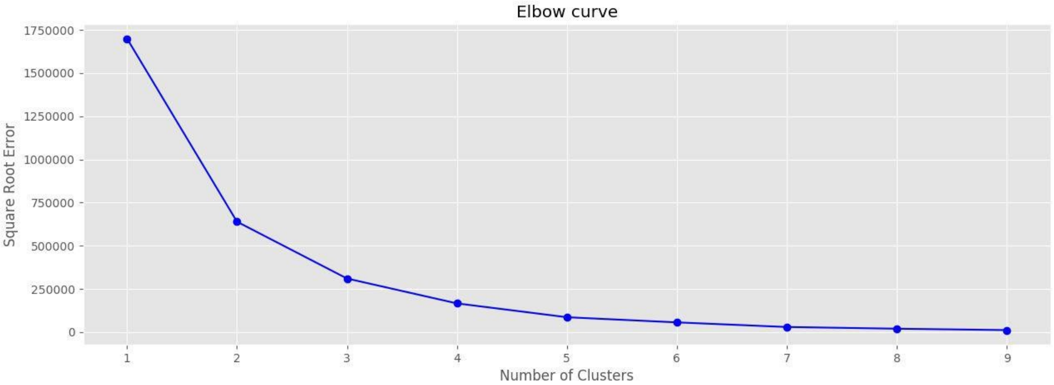
The Chi-squared statistic for candidate Hamilton is 3434.68

The Chi-squared statistic for candidate Madison is 1907.59

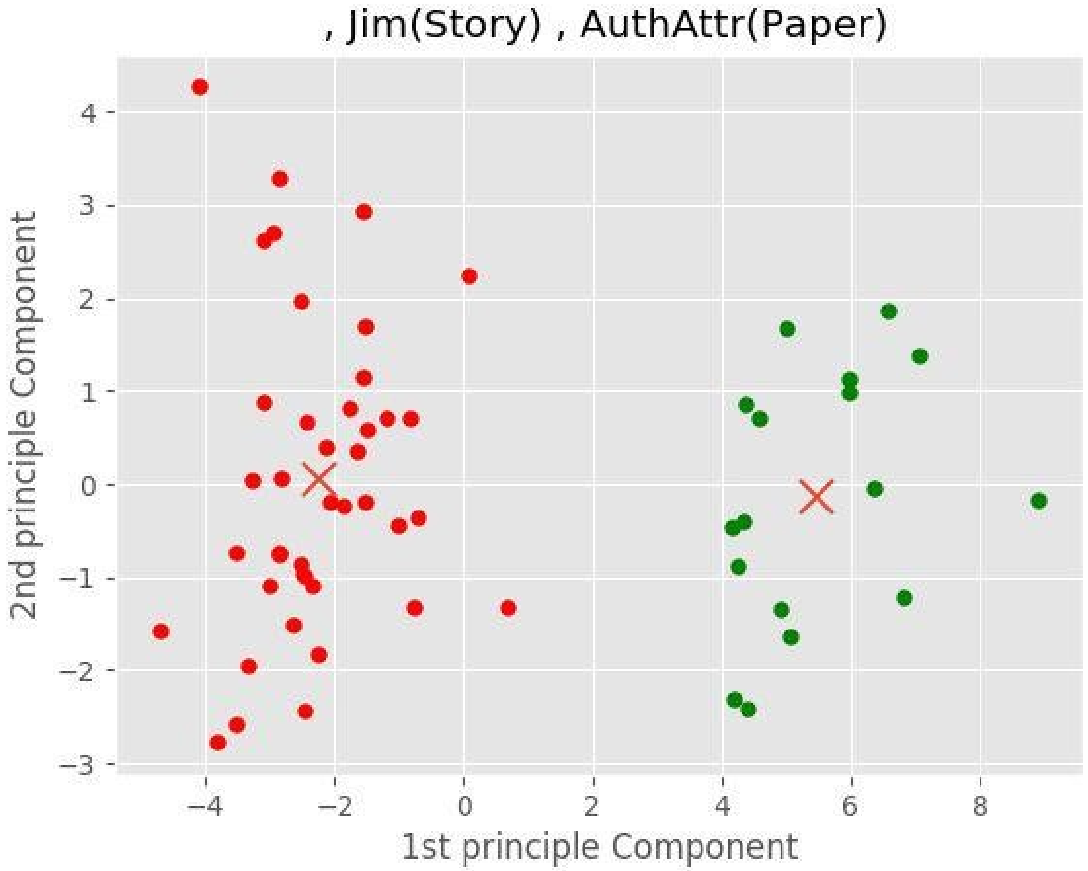
For a particular test case Federalist 64 essay:

- Delta score for candidate Hamilton is 1.768470453004334
- Delta score for candidate Madison is 1.6089724119682816
- Delta score for candidate Jay is 1.5345768956569326
- Delta score for candidate Disputed is 1.5371768107570636
- Delta score for candidate Shared is 1.846113566619675

4.3) Unsupervised Clustering of Documents



Caption: Elbow curve of square root error vs number of clusters



Caption: clustering of different writing styles

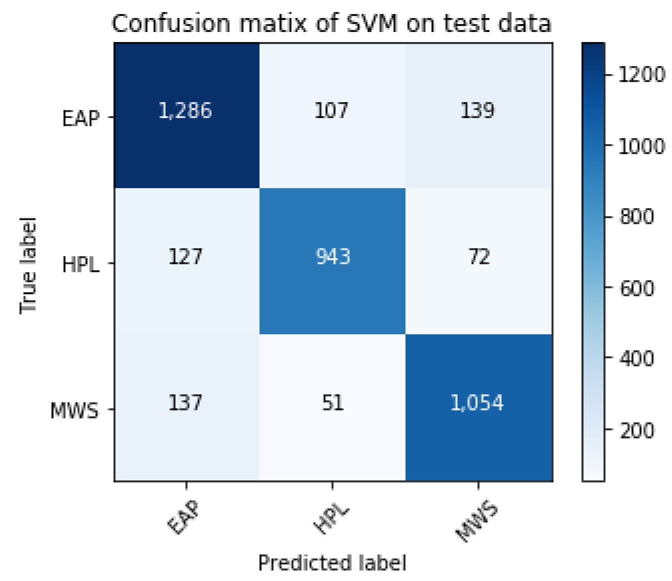
5 Analysis

5.1 Non Technical Documents

We see that the best input combination is Tokens + tf-idf + n-gram. Using SVM on this input, we get **83.83%** test accuracy. Using pos tagging significantly reduces the accuracy. SVM is the most successful algorithm for most of the feature inputs. GDBT performs well on character related inputs.

The best n-gram range is found to be 2 after hyperparameter tuning.

The below figure is the confusion matrix of the best system. All the authors are roughly equally confused with each other. The dataset is balanced and the performance on all the three authors is similar. This can also be seen from the fact that accuracy and f1 scores are close to each other.


Confusion matix of SVM on test data

True label \ Predicted label	EAP	HPL	MWS
EAP	1,286	107	139
HPL	127	943	72
MWS	137	51	1,054

Caption: Confusion matrix of the best system.

5.2 Technical Documents

As you can see from the graphs of **Mendenhall's Characteristic Curves** , the characteristic curve associated with the disputed papers looks like a compromise between Madison's and Hamilton's. The leftmost part of the disputed papers' graph, which accounts for the most frequent word lengths, looks a bit more similar to Madison's; the tail end of the graph, like Hamilton's. This is consistent with the historical observation that Madison and Hamilton had similar styles, but it does not help us much with our authorship attribution task. The best that we can say is that John Jay almost certainly did *not* write the disputed papers, because his curve looks nothing like the others.

As we can see from the chi square test, the chi-squared distance between the Disputed and Hamilton corpora is considerably larger than the distance between the Madison and Disputed corpora. This is a strong sign that, if a single author is responsible for the 12 papers in the Disputed corpus, that author is Madison rather than Hamilton.

As expected, Delta identifies John Jay as *Federalist 64*'s most likely author since delta score is low.

5.3 Unsupervised Clustering of Documents

From the elbow curve, we see that k =2 is the best trade off.

From the cluster graph, we see that our model has successfully found two clear and separable clusters of documents having different writing styles. This is an unsupervised method. If we know the author of a single document, we can assign it to all the documents of that cluster. More importantly, the algo only takes documents as inputs. This is very useful in case we don't have any clue of potential authors. The elbow method can tell us how many different authors are there.

6 Conclusion

In this project we detect the author of a document using stylometric features. We build methods for technical as well as non technical documents. We use supervised and unsupervised approaches to solve the problem. We can detect non technical documents using supervised learning (SVM) and a combination of lexical features with 83% accuracy. We predict the author of disputed technical documents using stylometric features and statistics. Finally, we show a stylometric method which can detect different writing styles in documents and cluster them based on the style. This method is very useful when we don't have potential authors or if we don't know how many different authors are present. Our methods can be used in industry for authorship attribution, solving disputer authorship claims, cluster documents etc.

7 Acknowledgement

We would like to acknowledge Perepu Satheesh Kumar for guiding and mentoring us on this project. We would also like to thank Dr. Anushree Bablani and Director sir for organizing and coordinating this course.

8 References

https://www.kaggle.com/christopher22/stylometry-identify-authors-by-sentence-structure/data
https://en.wikipedia.org/wiki/Tf%E2%80%93idf
https://en.wikipedia.org/wiki/Support-vector_machine
https://en.wikipedia.org/wiki/Logistic_regression
https://en.wikipedia.org/wiki/Gradient_boosting
https://en.wikipedia.org/wiki/Decision_tree
https://programminghistorian.org/en/lessons/introduction-to-stylometry-with-python#a-note-about-parts-of-speech
https://en.wikipedia.org/wiki/Chi-squared_test
https://en.wikipedia.org/wiki/Standard_score
https://ieeexplore.ieee.org/document/8981504
https://www.researchgate.net/publication/220435062_A_Survey_of_Modern_Authorship_Attribution_Methods

https://github.com/Hassaan-Elahi/Writing-Styles-Classification-Using-Stylometric-Analysis
http://homepage.divms.uiowa.edu/~mshafiq/files/shehroze-text-spinner-icdm2017.pdf
https://www.uni-weimar.de/medien/webis/publications/papers/stein_2011a.pdf
https://github.com/jpotts18/stylometry
https://link.springer.com/article/10.1023/B:CHUM.0000009225.28847.77
https://medium.com/@garykac/stylometry-machine-learning-pence-and-the-op-ed-bfb8041b5970

9 Appendix

Features Input	Algorithm	Accuracy	F1 score
Tokens	SVM	80.26	80.26
Tokens	LR	82.60	82.62
Tokens	GDBT	65.31	65.91
Tokens	DT	54.16	54.27
Tokens, tf-idf	SVM	83.57	83.58
Tokens, tf-idf	LR	80.32	80.36
Tokens, tf-idf	GDBT	65.71	65.19
Tokens, tf-idf	DT	51.65	51.73
Tokens, tf-idf, n-gram	SVM	83.83	83.83
Tokens, tf-idf, n-gram	LR	74.47	74.48
Tokens, tf-idf, n-gram	GDBT	64.24	64.80
Tokens, tf-idf, n-gram	DT	51.04	51.22
Chars	SVM	52.75	57.57
Chars	LR	56.89	57.67
Chars	GDBT	58.65	59.77
Chars	DT	45.09	45.12
Chars, tf-idf	SVM	56.92	57.62
Chars, tf-idf	LR	57.17	57.90
Chars, tf-idf	GDBT	58.40	59.13

Chars, tf-idf	DT	45.19	45.23
Chars, tf-idf, n-gram	SVM	69.25	69.29
chars, tf-idf, n-gram	LR	68.36	69.46
chars, ifidf, n-gram	GDBT	66.13	66.47
chars, tf-idf, n-gram	DT	46.67	47.64
tokens , postagging	LR	0.40	0.38
tokens , postagging	SVM	0.40	0.23
tokens , postagging	GDBT	0.40	0.23
tokens,postagging	DT	0.39	0.31

Caption: Results of all features and all algorithms.