When words are not enough

An evaluation of character n-grams and function words in author identification of musical artists

Alexander Nyström

Alexander Nyström
VT 2018
Examensarbete, 15 hp
Handledare: Pedher Johansson
Examinator: Marie Nordström
Kandidatprogrammet i datavetenskap, 180 hp

Abstract

When we write texts we unconsciously leave prints behind, these prints are things such as the words used, punctuation, special characters and more. There are several different approaches to author identification that utilises these features. All these methods have been applied to a variety of texts, everything from papers to poems, e-mail and forum posts. This study will use lyrics where the artists are the authors, on these the performance of two common features will be compared.

The two features that will get evaluated are character n-grams and function words. These are some of the most prominent features within author identification, where both have a track record of good performance. With high hopes for the performance the results showed that neither feature could reach the expected results. They were expected to achieve 70% and 65% accuracy respectively, however, the achieved average accuracy was only 40% and 35%. Even with the poor results some interesting finds were made. Some artists would have multiple band members write the songs which caused concern that it would affect the performance. Interestingly the results showed that multiple authors did not bad effect to the performance, in some cases they performed better than single authors.

Acknowledgements

I would like to express my gratitude towards my supervisor Pedher Johansson for the support and invaluable input throughout the work. I would also like to thank my friends and family for the support, with a special thank you to my friend Zander for helping me with the language.

Contents

1 Introduction

When we write we unconsciously create patterns with the words we choose as well as how we structure our text. This can be used as a basis for identifying who wrote a text. When an artist is writing a song, are these features apparent enough in the lyrics to identify their creator? Parts of this question will be explored in this work. Can we identify a artist by features in their writing style.

Author identification is a practice that has been around for a long time, but the first real successful case was Mosteller and Wallace [13] who identified the disputed authors of the federalist papers. Their results sparked life in the research field, and several methods for identifying authors have since been created and applied in different fields. The main approaches to identify authors are based on patterns in the language and structure of the text. Many of the patterns are things we as humans will use unconsciously and they will not change unless we actively try to mask them. Examples of patterns are: frequency of words, characters and sentences or length of words, sentences and paragraphs. Many of the patterns are not strong enough on their own and best results are achieved when combining the different features. Stamatatos [14] and Zheng et.al [17] provide detailed compilations of the different features used in the field.

This thesis aims to compare the performance of function words and character n-grams when identifying music artists using their lyrics. Character n-grams are overlapping character sequences that capture lexical, grammatical and orthographic preferences such as capitalisation, spelling and punctuation see Section 2.1 for a more in-depth description. Function words are words that do not carry any lexical meaning and acts as a form of glue in sentences, for a more detailed description see Section 2.2. Little work has been done to apply author identification methods on lyrics. However, lyrics has some properties that can be of interest. As an example artists usually write about multiple subjects and the structure of lyrics is different compared to other texts. An example of this is that lyrics tend to repeat sentences and sometimes paragraphs.

1.1 Related work

In *Neural Network Identification of Poets Using Letter Sequences*, Johan F. Hoorn et.al [10] identifies Dutch poets using letter sequences and machine learning. They use character n-grams and a window representation and evaluated the performance of the two features with three different machine learning methods. The corpus they used consisted of three different poets and 30 poems from each poet. A corpus is a collection of written text in this context it is a collection of written poems. Their aim was to identify which poem belong to which poet by first testing one against one followed by classifying all three at the same time. Even thought they did not use a large corpus they showed that it is possible to distinguish poets with character n-grams. This is important for this work since poetry and lyrics have a lot of similarities such as the wording to achieve rhythm and rhymes. Another important point is the success of using n-grams together with machine learning as a classifier, since this work aims to use machine learning as well.

Ying Zhao and Justin Zobel [16] evaluates how different classification methods perform using function words when identifying authors of news articles. They use different counts of authors from two to five authors, and they also evaluates the effect of number of works used per author in the classifiers. The results show that Bayesian networks are the most efficient classifier, and that function words can be used for author identification without the support of other features.

Zheng et al. [17] Makes a detailed compilation of related work and summarize the different features and methods used in the field of authorship analysis. The detailed compilation makes it easy to find work of similar nature. They also propose a framework for identification of authors of online messages, where they utilize several features divided into four groups: lexical, syntactic, content-specific, and structural features. A simplified list of the features used: The lexical features are based on character counts and frequency, as well as word length, count and frequencies. Syntactic features focus on frequency of punctuation and function words. Content-specific are words that are common in online messages. Structural features are number of lines, sentences and paragraphs. Structural features includes also the number of words, sentences and characters per paragraph. Their test data consisted of sale messages from a site for online trading, they collected 20 authors with average of 48 messages each, and an average word count of 169 per message. They also collected Chinese messages. However, they were not sale messages but a collection of different topics. The Chinese corpus consists of 20 authors and 37 messages and 807 average word count. Then they trained three different machine learning classifiers these classifiers were a C4.5 decision tree, neural network and support vector machine. The support vector machine outperformed the others with neural network being second. This article gives evidence that it is possible to identify a decent number of authors with limited texts and achieve an accuracy of ~80% . The article also show the power of support vector machines and neural networks.

De Vel et al.[7] Uses a selection of style features together with a support vector machine to identify e-mail authors. The corpus used consisted of 3 authors of 156 e-mails with an average word count of 76 on three different categories. The presented results show that it is possible with limited amount of text to identify authors using a support vector machine. However, they only used 3 authors which is a rather low count, and it would be more interesting with a corpus consisting of a larger count of authors. None the less, it brings forth evidence of the validity of support vector machines and function words.

In the unpublished work by Corney et al. [6], they used a selection of style features to identify e-mail authors using a support vector machine. This time they evaluated the impact of word count as well as the performance of some of the features. In this paper they use character n-grams (n=2) which performs good in their initial tests, but they do not test it on the e-mails. Their results show that function words are a good feature to distinguish authors, and that character n-grams may find content based features and thus not only discriminate on style alone.

1.2 Expected results

Based on results from previous works some expectations can be laid out. It can be expected that character n-grams will prove to perform better than function words when comparing the two methods. This expectation is based on works where character n-grams have been shown to be one of the better author identification methods on opinion columns from newspapers, online messages, e-mails and books [9, 12]. Looking at the performance of the individual methods it is difficult to predict how it will perform since lyrics have some differences from other texts. On dutch poets n-grams have achieved an accuracy of around 70-78% with three authors [10]. How the performance will be with more than three authors and another type of text is difficult to predict. It is expected to have slightly lower performance than it has shown on other types of texts, it is then expected to be lower than 70%. The performance for function words are also an uncertainty since it most commonly have been combined with other features. However, based on the study performed by Koppel et.al [12] the performance on e-mail and blog posts was around 63% when using a support vector machine implementation. Given that e-mail and blog posts are also limited in length it could be expected that function words would perform slightly lower than 63% for four or more authors.

2 Features and classification method

This work will focus on the performance of two features for author identification which are character n-grams and function words. In this section we will present these features as well as the method used for classification and how performance will be measured.

2.1 Character n-grams

Character n-grams are short overlapping sequences of characters of length n. An example the beginning of this section in the form of tri-grams which are three character n-grams are,
| Cha | har | ara | rac | act | cte | ter | er_ | r_n | _n- | n-g | -gr | gra | ram | ams |
and so on (where "_" indicates space). These n-grams capture lexical preferences as well as grammatical and orthographic preferences, that includes capitalization, punctuation and white spaces. However, they may also capture content specific features [12, 6]. When applying this feature for author identification the frequency of occurrence is calculated for all n-grams in the text. This technique has been used with great results by many researchers. They have shown that with variety of machine learning methods n-grams can achieve 65-88% accuracy for identifying up to four authors. [9, 10, 11, 12].

2.2 Function words

Function words are an important part of the structure of a sentence. They are words that do not carry any lexical meaning, their purpose is like a glue that holds the sentences together. Example of function words are *the, in, about* and so on. Because function words are separated from content they are ideal for identifying authors writing about several different subjects. Other important features of function words are that we use them unconsciously when writing [14]. Like with n-grams the frequency of occurrence is used when applied to author identification. In previous work function words have usually been used together with other features to discriminate authors [17, 7]. However, in some work it has been used as a stand alone feature [16, 9, 6, 12]. In Appendix A.2 a list of the function words used can be found, the selection of words came from the work by Zheng et al. [17].

2.3 Machine learning algorithm

A common machine learning method in the field of author identification is Support Vector Machine (SVM) and has been used with good results [8, 7, 12, 17]. SVM are a linear classifier that uses a hyperplane to separate values that belongs to different classes. Figure 1 displays simple example with a hyperplane separating two classes and the dotted lines are the margins and the class samples that are on the margins are called support vectors. What the SVM does is that it clusters the values of the classes in this context a class is an author. The two groups are then linearly separated with a function with as much space as possible this is seen in Figure 1. If the classes are not linearly separable the SVM will move the values into a higher dimensional space and then separate them by a hyperplane. Support vector machines are a binary classifier that is they can only separate two classes. When it is required to classify several classes one uses a method called one versus all. One versus all uses one SVM for each class and then trains it to classify that class as positive and all other classes as negative.

In this work the scikitlearn python library has been used which provides several implementations of different machine learning methods. From this library the linear support vector classifier (LSVC) has been used [5]. When training the SVM a common method used is cross-validation, one of the more common ones are k-fold cross-validation [17, 12, 7, 16, 6]. K-fold cross-validation splits the data into k separate subsets and then trains the method on all but one of the subsets and uses the last for validation.

2.4 Performance measures

When measuring the performance of binary classifiers there are a few measures available. Precision is the measure of probability that a true guess genuinely belongs to that class and is calculated as in Equation 1. Next measure is recall which gives the number of objects found for that class and it is calculated as in Equation 2. With the recall and precision one can calculate the F1 measure which gives the retrieval performance and F1 is calculated as in Equation 3[15]. The last measure is the overall accuracy which gives the accuracy over all classes, and it is calculated as in Equation 4 where C is correct guesses and G is all guesses.

tp = true positive (True positive are those that are correctly classified as positive)
fp = false positive (False positive are those that are incorrectly classified as positive)

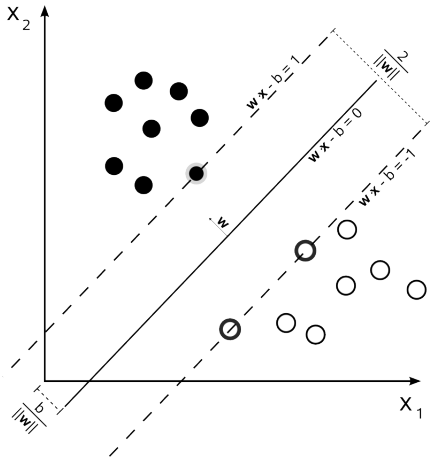

Figure 1: A support vector machine with a maximum-margin hyperplane that separates two classes, black and white. (Source [3])

fn = false negative (False negative are those that are incorrectly classified as negative)
tn = true negative (True negative are those that are correctly classified as negative)

$$P = \frac{tp}{tp + fp} \tag{1}$$

$$R = \frac{tp}{tp + fn} \tag{2}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{3}$$

$$Acc = \frac{C}{G} \tag{4}$$

3 Performing of experiment

This section will cover the outlay and performing of the experiment as well as the data used to perform the experiment.

3.1 Dataset

This experiment required a great collection of artists and their lyrics. Such a dataset was available on Kaggle [2] which is a website for sharing open datasets. The dataset that was found was called “Every song you have heard (Almost)!” [1]. This dataset consists of over 500’000 song lyrics and their artists. However, only a limited subset of these were used in the experiment.

For the best chance at good performance some requirements was placed on the data. These requirements were the amount of available lyrics and the writer of the lyrics had to be a member of the band or the artist them self. The required song count will help the SVM to

find good classification without overfitting. The other requirement that the lyricist had to be a member of the band or the artist them self will reduce the chance of miss classification. If two artists would share lyricist there is a chance they would be classified as one or the other. Something important to note is to get a adequate amount of artists an exception was made in case of band members writing the lyrics. This exception is that bands where multiple members have written the lyrics are allowed, if this had any affect on the performance will be discussed in Section 5.

From the dataset the first collection of songs and artists were used, and filtered by number of songs. The filter was set at 100 songs minimum, so all artists who had less than 100 songs in the collection got removed. After the first step of filtering roughly 360 artists remained with 100+ songs. Next step was to clean the song names from special markings found in the dataset, as well as removing duplicates and live or remix versions. Important to note is that not all duplicates could be removed automatically and there could still be some duplicates in the dataset and some of the live versions were kept if no other version of that song was available. It is not expected to affect the results in a significant manner. The worst case is that it will have the songs both in the training and test set and thus have it easier to classify that specific song. The next filter step was to remove songs not written by the artist or band members. To accomplish this Wikipedia was scraped for band members both current and previous and Musicbrainz [4] database was used to query song writers. The last count of artists was 189. However, after the filtering not all of them had at least 100 songs. So the final batch of artists got filtered once again to remove those who did not have 100+ songs. The list of the final artists can be seen in Appendix A.1.

3.2 Data format

To prepare the data for the machine learning the frequencies for the n-grams and function words were collected into vectors from all songs. These frequencies are values between 0 and 1 that shows how frequent a specific n-gram or function word is within a song. For the character n-grams only the 30 most frequent n-grams in each song are used as features, this gives all authors an equal feature impact. In the work by Hoorn et.al the 30 most frequent n-grams was used when identifying poets [10] this was the inspiration to use the 30 n-gram limit. For the function words the 20 most frequent words in each song are used which was the highest number of words possible without having any word with a frequency at 0%. Class labels are represented by integers where each artist is assigned a value and they are stored in a list with the same index as their songs. The integer can be converted to a name using the values as index in a list containing artist names.

3.3 Training and Testing

Before the SVM can be trained the formatted data is split into subset containing only four classes which are then split into a training and test set with a 80/20 ratio. The training set is then fed into the SVM and to find the optimal settings an exhaustive search using 30-fold cross-validation was used. When the training is completed the estimator is tested on the test-set and from the results the performance measures are calculated (Section 2.4) and a confusion matrix is generated. This was done for all combinations of artists to get good coverage of the performance of n-grams and function words.

4 Results

The complete collection of results is not feasible to present since all combinations of artists have been tested. Instead to get a good insight in the performance as a whole for both methods the results that will be presented are average values for each artist as well as an overall average. This will not only give good insight in the performance but also shed some light on some interesting behaviours for the individual artists. In Table 1 the average results can be viewed with n-grams and function words side by side to ease the comparison of the two methods. The columns show the different performance metrics explained in Section 2.4.

Table 1 Average results for individual artists and average values for the four different measures.

	N-grams			Function Words		
Artist	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Black Sabbath	0.51	0.43	0.37	0.448	0.392	0.363
XTC	0.359	0.328	0.309	0.271	0.353	0.295
Clutch	0.27	0.395	0.269	0.376	0.452	0.394
The Rolling Stones	0.43	0.47	0.455	0.363	0.297	0.294
U2	0.4	0.26	0.3	0.3	0.138	0.166
The Beatles	0.503	0.539	0.5	0.398	0.56	0.462
Electric Light Orchestra	0.368	0,223	0.253	0.376	0.403	0.383
Joni Mitchell	0.444	0.649	0.496	0.351	0.493	0.39
R.E.M.	0.362	0.294	0.315	0.221	0.174	0.186
Megadeth	0.282	0.328	0.274	0.382	0.25	0.285
The Kinks	0.398	0.454	0.396	0.347	0.229	0.267
Bad Religion	0.393	0.592	0.453	0.347	0.562	0.417
Green Day	0.24	0.174	0.185	0.371	0.478	0.414
Jethro Tull	0.52	0.391	0.388	0.365	0.255	0.272
Avg/total	0.391	0.462	0.354	0.351	0.359	0.327
	Accuracy: 0.399			Accuracy 0.349		

To add some interesting discussion in addition to the main focus of performance comparison. The results from the test where both methods performed the best can be seen in Table 2. Another way to view these results are in the confusion matrices in Figure 2 which gives a visualization of how the predictions for both methods were distributed. The confusion matrix is a good tool to closer analyse the behaviour of the classifier, it makes it easier to view if some authors often gets mixed up. This test was performed on the same artist but the distribution of songs are not the same. However, this is meant to showcase when both methods are at their best and not as a base for the main comparison.

5 Discussion

To answer the aim of this work the total average values of the performance measures in Table 1 are the base for the comparison between the two methods. In the table it is clear that n-grams performs better on all measures than function words. The results display that there is not a large difference in performance between the two methods, the difference is just a few

Table 2 Results from the tests that generated the best performance for both n-grams and function words.

	N-grams			Function Words		
Artist	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
The Beatles	0.58	0.73	0.64	0.39	0.57	0.46
Jethro Tull	0.48	0.46	0.47	0.35	0.25	0.29
Clutch	0.60	0.33	0.43	0.52	0.47	0.49
Green Day	0.41	0.52	0.46	0.48	0.48	0.48
Avg/total	0.52	0.51	0.50	0.44	0.44	0.43 1
	Accuracy: 0.51			Accuracy 0.44		

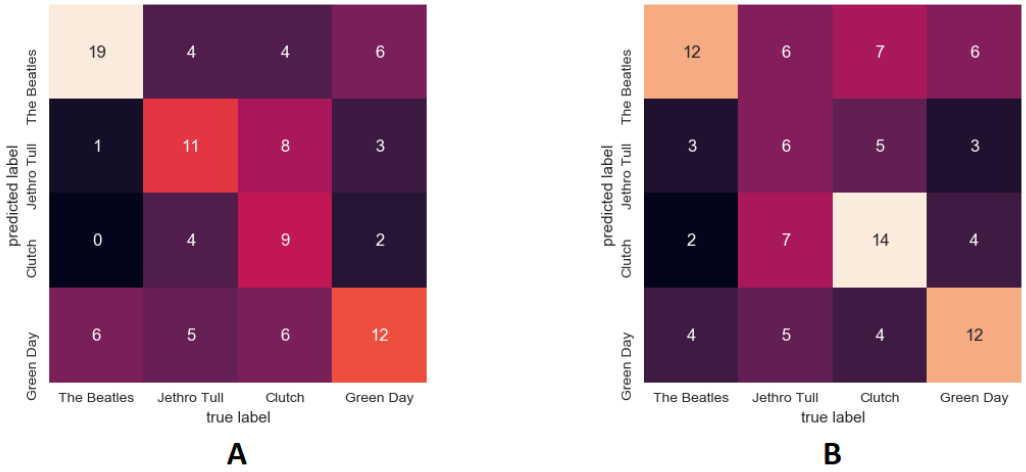

Figure 2: Confusion matrices for both n-grams (**A**) and function words (**B**). The numbers are real numbers, that is each number is the number of songs predicted to belong to that artist.

percentage points. However, it is clear to say that of the two methods n-grams have it easier to identify authors of lyrics than function words. Which was one of the predictions laid out in Section 1.2. When evaluating the performance of the two methods we compare the performance against what was expected. None of the methods could achieve the expected performance. It was expected that n-grams would perform with an accuracy closer to 0.7 but the average is closer to 0.4 in accuracy. The expectation on function words was to have an accuracy close to 0.63 however, the average accuracy was closer to 0.35. It should be said that even with the poor performance both methods achieve a average score that is higher than just blindly guessing, which would have been at 0.25. This implies that both methods can identify authors. However, the performance is too low to be useful in a real scenario. Taking a closer look at the precision and recall for function words, the two measures are close with barely a percentage point better recall than precision. That indicates that function words are just as good at predicting correct as it is at finding the available songs for the artists. For character n-grams on the other hand recall is better which suggests it is better at finding all the songs for the artists but not as precise with the predictions. However, the precision for n-grams is better than function words so it is more precise and has better recall than function words.

Looking at the individual artists most of them are easier to identify using n-grams than with function words. There are only four artists that are easier to identify using function words, many of which are those with bad performance using n-grams. Green Day for example is below chance level using n-grams but for function words it is one of the better performing artists. The same can be seen for Clutch which is just above chance level using n-grams but displays good performance using function words. Green Day has mainly one writer credited for the lyrics which could indicate that the use of function words will more easily portray the writer. On the other hand many of the Clutch songs are credited to the band, which would indicate that all members are behind the lyrics. However, it could still be just one writer and the rest giving input. Without further investigation it is difficult to answer if it is something in their style of writing that allows for the use of more function words, or if the answer lies in the authors.

An interesting observation can be made of the best performing artists Black Sabbath, The Beatles and The Rolling Stones. These three artists have multiple members credited for lyrical writing. However, in all three cases there are one or more member that appears as an writer for almost all songs. This proves that even with the relaxed requirement on the writers allowing some to have multiple writers, it did not affect the performance in a bad way. Another interesting observation is that some of the artists which only has one writer actually performs worse than some with multiple writers. Green Day is one of those examples they do perform good for function words however, The Beatles have slightly better performance with an average of two lyrical writers.

The results from the best case displayed in Table 2 is not a base for the main comparison. However, it is interesting to discuss how the two methods perform when they are in the best suited environment. Most of the artists have an performance quite a bit above their average values. Especially n-grams and Clutch shows a precision of 0.6 when the average is 0.27. This shows that in a good setting the performance will be able increase significantly above average. The same pattern as in the average result can be viewed for Green Day which results in a better performance using function words. In this case the difference between n-grams and function words on Green Day is not as great as it is on average. The average total from this test is in the range of ten percentage points higher than the average across all tests. This is a significant increase, thus it would not be unfeasible to reach results closer to the expected in a perfect scenario. However, it is far less feasible across all artist combinations.

When observing the confusion matrices Figure 2 shows that Jethro Tull is spread out across all authors for both methods. For function words most Jethro Tull songs are not labelled Jethro Tull but instead labelled Clutch. The average results showed that Jethor Tull was one of the worst performing for function words and that manifests clearly in the confusion matrix. For n-grams Jethro Tull is not the worst performing in the average results but in the confusion matrix it is clear that quite a few songs are miss classified. However, in both cases the score seen in Table 2 tells that they both are above average. This is a clear indication to why none of these methods perform at a level which can be used in a useful scenario. Even with some decently high scores the predictions are still spread across the authors.

6 Conclusion

The results in this work did not meet the expectations placed. The performance was significantly lower than expected. However, one of the expectations was proven true, that character n-grams would perform better than function words. Where in the average result only four out of fourteen artists were easier to identify using function words, among them Green Day which had an appalling performance with n-grams but one of the better with function words. But the difference in performance was not as great as expected but significant enough to conclude that n-grams are the better approach. With this poor performance one can conclude that none of the methods are good enough at identifying artists from lyrics for any real use-case. If one were to get the accuracy above 0.7 it could be used as an automatic author tagger in music databases to credit the author, other uses could be to discern if a song really belong to a band or artist.

Another conclusion is that it does not matter if multiple band members are credited for songs if one or more of them are credited in most songs. This was an interesting find that even thought multiple people was involved in writing they proved to be just as easy to identify as those with only one author. Some even out performed the artists with only one author, namely The Beatles and Black Sabbath.

Except the performance n-grams have more benefits such as that it is easier to collect a good amount of data. Since n-grams only require character combinations to appear frequent it is easier to find than function words which require whole words to appear frequent. Especially given the limited format of lyrics, if the lyrics were longer there is a chance it would affect the performance in a positive manner for function words.

7 Future work

There are hints at that n-grams could pick up on content specific features of a text [12, 6]. To investigate if n-grams really can pick up on content specific features lyrics could be a good text type to analyse. Since the same author could write about a variety of subjects but there can also be similarities between the subject and content from other artists.

From a language and writing stand point it could be interesting to study the lyrics of the artists that was easy or difficult to identify. By doing so investigate if there is anything obvious in the lyrics that would prove to affect the results, this could also hint at what type of features one should use when identifying authors of lyrics.

Given that in this thesis n-grams had more features than function words, it would be interesting to investigate if the number of features used affects the performance in a significant manner. Moreover it would be interesting to find the maximum performance possible to achieve, by selecting the most promising artists and test other machine learning approaches and compared their performance to the SVM. It would be interesting to test a neural network to see if it can achieve better results than the SVM. Trying other features would also be an interesting angle since neither n-grams or function words could achieve expected results. It would also be possible to combine the features and see if the combination of function words and n-grams could perform better than as separate features. In addition to combining n-grams and function words it would be interesting to combine more features, to see if a weighted combination of several features could reach expected results.

References

[1] Every song you have heard (almost)! https://www.kaggle.com/artimous/every-song-you-have-heard-almost. Accessed: 2018-04-03.

[2] Kaggle datasets. https://www.kaggle.com/datasets. Accessed: 2018-04-03.

[3] Maximum-margin hyperplane. https://en.wikipedia.org/wiki/Support_vector_machine#/media/File:Svm_max_sep_hyperplane_with_margin.png. Accessed: 2018-04-12.

[4] Musicbrainz. https://musicbrainz.org/doc/MusicBrainz_Database. Accessed: 2018-04-17.

[5] Scikit-learn: Linearsvc. http://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html#sklearn.svm.LinearSVC. Accessed: 2018-05-1.

[6] Malcolm W Corney, Alison M Anderson, George M Mohay, and Olivier de Vel. Identifying the authors of suspect email. 2001.

[7] Olivier De Vel, Alison Anderson, Malcolm Corney, and George Mohay. Mining e-mail content for author identification forensics. *ACM Sigmod Record*, 30(4):55–64, 2001.

[8] Joachim Diederich, Jörg Kindermann, Edda Leopold, and Gerhard Paass. Authorship attribution with support vector machines. *Applied intelligence*, 19(1-2):109–123, 2003.

[9] Jack Grieve. Quantitative authorship attribution: An evaluation of techniques. *Literary and linguistic computing*, 22(3):251–270, 2007.

[10] Johan F Hoorn, Stefan L Frank, Wojtek Kowalczyk, and Floor van Der Ham. Neural network identification of poets using letter sequences. *Literary and Linguistic Computing*, 14(3):311–338, 1999.

[11] Bradley Kjell. Authorship determination using letter pair frequency features with neural network classifiers. *Literary and Linguistic Computing*, 9(2):119–124, 1994.

[12] Moshe Koppel, Jonathan Schler, and Shlomo Argamon. Computational methods in authorship attribution. *Journal of the Association for Information Science and Technology*, 60(1):9–26, 2009.

[13] Frederick Mosteller and David Wallace. *Inference and disputed authorship: The Federalist*. Addison-Wesley, 1964.

[14] Efstathios Stamatatos. A survey of modern authorship attribution methods. *Journal of the Association for Information Science and Technology*, 60(3):538–556, 2009.

[15] Yiming Yang. An evaluation of statistical approaches to text categorization. *Information retrieval*, 1(1-2):69–90, 1999.

[16] Ying Zhao and Justin Zobel. Effective and scalable authorship attribution using function words. In *Asia Information Retrieval Symposium*, pages 174–189. Springer, 2005.

[17] Rong Zheng, Jiexun Li, Hsinchun Chen, and Zan Huang. A framework for authorship identification of online messages: Writing-style features and classification techniques. *Journal of the Association for Information Science and Technology*, 57(3):378–393, 2006.

A Appendix

A.1 Artists

The Rolling Stones	Joni Mitchell	U2	Bad Religion	Megadeth
Electric Light Orchestra	XTC	The Beatles	The Kinks	R.E.M.
Green Day	Jethro Tull	Black Sabbath	Clutch	

A.2 Function words

a	between	in	nor	some	upon	about
both	including	nothing	somebody	us	above	but
inside	of	someone	used	after	by	into
off	something	via	all	can	is	on
such	we	although	cos	it	once	than
what	am	do	its	one	that	whatever
among	down	latter	onto	the	when	an
each	less	opposite	their	where	and	either
like	or	them	whether	another	enough	little
our	these	which	any	every	lots	outside
they	while	anybody	everybody	many	over	this
who	anyone	everyone	me	own	those	whoever
anything	everything	more	past	though	whom	are
few	most	per	through	whose	around	following
much	plenty	till	will	as	for	must
plus	to	with	at	from	my	regarding
toward	within	be	have	near	same	towards
without	because	he	need	several	under	worth
before	her	neither	she	unless	would	behind
him	no	should	unlike	yes	below	i
nobody	since	until	you	beside	if	none
so	up	your				