Lyrics Classification using Naive Bayes

Dalibor Bužić [*], Jasminka Dobša [**]
[*] College for Information Technologies, Klaićeva 7, Zagreb, Croatia
[**] Faculty of Organization and Informatics, Pavlinska 2, Varaždin, Croatia
dalibor.buzic@vsite.hr, jasminka.dobsa@foi.hr

Abstract - **Text classification is an important and common task in supervised machine learning. The Naive Bayes Classifier is a popular algorithm that can be used for this purpose. The goal of our research was prediction of song performer using Naive Bayes classification algorithm based solely on lyrics. A dataset that has been created consists of lyrics performed by Nirvana and Metallica, 207 songs in total. Model evaluation measures showed very good results: precision of 0.93, recall of 0.95 and F_1-measure of 0.94, therefore lyrics classification using Naive Bayes can be considered as successful.**

Keywords - Naive Bayes classifier, text classification, machine learning

I. INTRODUCTION

Text classification is an important and common task in supervised machine learning. Its application is in email spam detection, sentiment analysis, language detection of written text, classification etc. Many classifiers can be used for document classification. Some of them are neural networks, support vector machines, genetic algorithms, Naive Bayes classifier, k-nearest neighbours and Rocchio classifier [1].

The quantity of music, especially on the internet, is growing rapidly and its organizing is a challenging task. Given the huge size of music collections, classification of music should be made automatically. Classification can be made according to genre, mood, performer, geographical region, etc.

To make classification successful, one can rely on audio features such as tempo, rhythm, timbre, pitch, loudness or lyric features such as word and sentence length, word frequencies, word n-grams, sentence and phrase structure, errors, synonyms, rhyme patterns etc. According to [2] most existing work on automatic music mood classification is based on audio features (spectral and rhythmic features are the most popular).

Depending on type of classification, combining audio and lyrics information is a common approach.

In [3] four very distinct genres (classical, jazz, metal and pop) were chosen for audio-based classification using Mel Frequency Cepstral Coefficients. Accuracy in genre prediction when Direct Acyclic Graph Support Vector Machines was applied varied from 67 % to 97 %. When Neural Networks were used, accuracy varied from 76 % to 100 % depending on genre.

Automatic identification of music performers, given a set of piano performances of the same piece of music is an interesting research described in [4]. Pianists played two pieces by Frederick Chopin. Success rate was high: the accuracy was 70 % in 10-class task.

Fell and Sporleder in [5] dealt with problem of finding out whether it is possible to automatically predict the approximate publication time of a song given its lyrics. They chose pop/rock songs and divided them into three periods: 2008 and newer, from 1998 to 2001, and those published before 1988. Results showed that songs which are published 20 years and more ago can be distinguished relatively well, but for newer songs results of classification are relatively low.

Authors in [6] report that there is no significant difference in results of music mood classification depending on whether stemming was used or not. In [7] authors highlight that stemming and removing of stop words may do more harm than good when dealing with multilingual lyrics.

Text authorship identification is a field with long research history [8]. The main idea behind statistically or computationally supported authorship attribution (which started at the end of 19th century) is that the texts written by different authors can be distinguished by measuring some textual features [9]. This field rapidly evolved with the development of machine learning classification techniques.

The goal of this research was testing whether the Naive Bayes classifier can successfully predict song performer based solely on lyrics. A dataset consisting of lyrics of two performers (Nirvana and Metallica) was created for this purpose. Two performers are chosen deliberately to separate problems of classification according to performer from the problem of classification according to the genre of music because genres of their music are not far away from each other. Nirvana is a rock band, while Metallica is heavy metal (which is one sub genre of rock) band. No single author writes lyrics for one performer, but songs are written having a performer in mind (and audience of course), so style and genre of songs should be close to each other. As a matter of a fact, sometimes one song is written by more than one author. In the case of Metallica, many songs are written by three or four authors.

As dataset has 127 Metallica's and 80 Nirvana's songs, Naive Bayes Classifier was used, because it is suitable for small datasets [10].

The remainder of this paper is organized as follows. In the next section we briefly describe the methods and measures we used. In Section 3 we describe our experiment and present the results. In Section 4 we draw conclusions and point out future directions.

II. METHOD AND MEASURES

A. *Naive Bayes*

Naive Bayes is a machine learning algorithm whose classification efficiency is proved in applications such as document categorization and e-mail spam filtering [11]. This classifier learns through a document classification algorithm, and is based on a simple usage of the Bayes' rule [12]:

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)} \tag{1}$$

wherein:

- c is a class,
- d is a document,
- $P(c)$ is a class probability,
- $P(d)$ is the probability of a document,
- $P(d|c)$ is conditional probability of the class for the given document d,
- $P(c|d)$ is conditional probability that document d belongs to class c.

Naive Bayes classifier is characterized by: [13]

- computational efficiency,
- low variance,
- incremental learning,
- direct prediction of posterior probability,
- robustness to noise and
- robustness on missing values.

Computational efficiency in modeling and predicting is an unquestionable advantage over some other classification algorithms, which is due to the possibility of easy parallelization, especially important for large datasets. To fore mentioned characteristics it is valuable to add two more: resistance to overfitting and ability of handling with large number of attributes without need their selection [14].

B. *Performance measures*

After creating a machine learning model, it is necessary to measure model performance to decide if the model is satisfactory, whether it can be improved or even discarded. Model should make as low mistakes as possible - but the concept of mistake can be defined on different ways, depending also on the problem domain. Below are some of the most common evaluation measures. They originate from the confusion matrix (Table 1) which contains the classifier's decisions in the rows, and the actual decision about classification in the class in the columns. The four fields of the table contain number of true positive (TP), true negative (TN), false positive (FP) and false negative (FN) classified documents.

Table 1. Confusion matrix

		actual	
		YES	NO
predicted	YES	TP	FP
	NO	FN	TN

Precision is expressed as the proportion of positive cases that are correctly recognized as positive over all cases classified as positive and is calculated according to the formula:

$$\text{precision} = \text{TP} / (\text{TP} + \text{FP}) \tag{2}$$

Recall is expressed as the proportion of positive cases that are correctly recognized as positive over all actual positive cases and is calculated according to the formula:

$$\text{recall} = \text{TP} / (\text{TP} + \text{FN}) \tag{3}$$

Accuracy is expressed as the proportion of correctly classified cases over all cases and is calculated according to formula:

$$\text{accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{4}$$

Error is expressed as the proportion of incorrectly classified cases over all cases and is calculated according to formula:

$$\text{error} = (\text{FP} + \text{FN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{5}$$

or simpler:

$$\text{error} = 1 - \text{precision} \tag{6}$$

Individual measures should not be considered separately. It would be easy to construct a completely useless classifier which would classify all cases as positive, making the recall measure perfect 1. Precision and recall are complementary, as one represents the ability to detect positive cases, and the other ability to avoid incorrect detection of negative cases. By increasing one measure it is likely to decrease another (or, at best, another will remain the same) [14].

A measure that combines precision and recall is called the F_1-measure and represents their weighted harmonic mean. It is calculated according to the formula:

$$F_1 = ((2 \times \text{precision} \times \text{recall}) / (\text{precision} + \text{recall}) \tag{7}$$

The F_1-measure is one of the most commonly used single-number measures in information retrieval, natural language processing and machine learning. It is worth mentioning that this measure has more practical issues, some of which are: [15]

- like precision, recall and accuracy, it is also focused only on one class,
- like precision, recall and accuracy, it is also biased towards a dominant class,
- it does not consider true negative (TN) cases and

- it assumes that the actual and the prognosed distributions are equal.

The last evaluation measure to be mentioned here is the ROC (Receiver Operating Characteristic) curve. It is a graphical representation (Figure 1) of the binary classifier performance on which the curve represents a compromise between true positive and false positive cases.

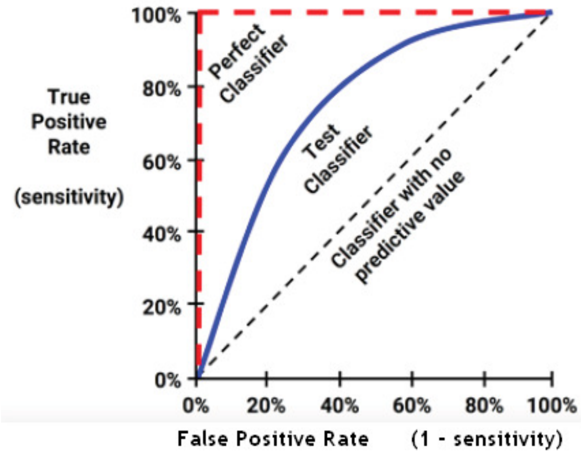


Figure 1. ROC curve (source: [10])

The black line on the diagonal represents a classifier that true positive and false positive cases detects at the same rate, therefore is not useful in the classification. In the contrast, a perfect classifier marked with red line predicts 100 % true positive with 0 % false negative cases. What is the curve of the actual classifier closer to the red, it is better for detecting positive cases.

III. EXPERIMENT AND RESULTS

The goal of this research was to find out if the selected classifier can correctly identify the performer (Metallica or Nirvana) only by lyrics. A set of data was made for the purpose of research, and the data was prepared for processing. Subsequently, the model was trained and evaluated. The last step was model improvement.

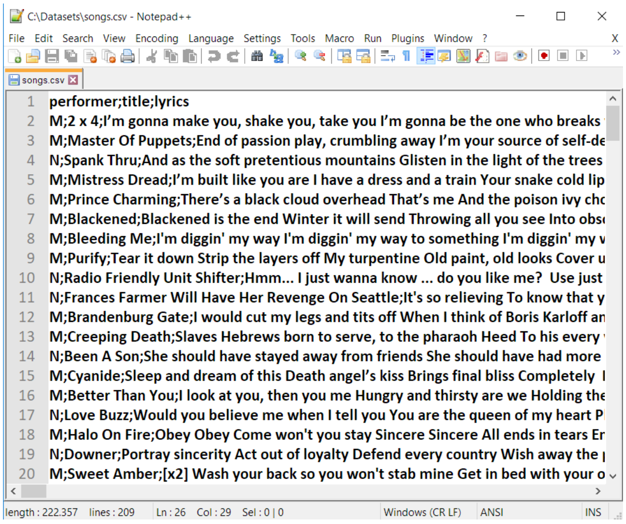
A. *Data collection and preprocessing*

The research question in this paper was whether a classifier (and to what extent) based only on lyrics can recognize whether it is a song of Nirvana or Metallica.

The first step was creation of a dataset. All songs were obtained from azlyrics.com website. Dataset consists of three columns (type, title and song lyrics) and 207 rows (127 Metallica's and 80 Nirvana's songs). The first column, *type*, contains one-letter information about to whom the song belongs ('M' for Metallica or 'N' for Nirvana). The second column, *title*, contains song titles – it was not used in the research, but it is important to easily recognize particular song and control possible duplicates. The third column, *lyrics*, contains song lyrics. In some cases it was not entirely clear whether particular song actually belongs to observed band (due to the music career of band's frontman before establishing band, for example). To resolve such issues, only songs listed on Wikipedia's pages (precisely https://en.wikipedia.org/wiki/List_of_songs_recorded_by_Metallica and https://en.wikipedia.org/wiki/List_of_songs_recorded_by_Nirvana) could be included into the dataset. It is also important to emphasize that dataset does not represent entire discography of two bands.

After initial dataset creation, randomization of rows was made. Since at the end of dataset creation there was a known share of songs (61,4 % Metallica and 38,6 % Nirvana), and the fact that two-thirds of data (138) would be used for learning and the remaining third for testing, in the training set proportional number of both band's songs was placed (85 Metallica's and 53 Nirvana's songs).

At the end, once again set for training and set for testing were separately randomized. This adjustment later enabled simplifying training and testing operations in the R tool. Part of the final dataset is shown in Figure 2.



C:\Datasets\songs.csv - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
songs.csv

```
performer;title;lyrics
M;2 x 4;I’m gonna make you, shake you, take you I’m gonna be the one who breaks
M;Master Of Puppets;End of passion play, crumbling away I’m your source of self-de
N;Spank Thru;And as the soft pretentious mountains Glisten in the light of the trees
M;Mistress Dread;I’m built like you are I have a dress and a train Your snake cold lip
M;Prince Charming;There’s a black cloud overhead That’s me And the poison ivy chc
M;Blackened;Blackened is the end Winter it will send Throwing all you see Into obs
M;Bleeding Me;I'm diggin' my way I'm diggin' my way to something I'm diggin' my v
M;Purify;Tear it down Strip the layers off My turpentine Old paint, old looks Cover u
N;Radio Friendly Unit Shifter;Hmm... I just wanna know ... do you like me?  Use just
N;Frances Farmer Will Have Her Revenge On Seattle;It's so relieving To know that y
M;Brandenburg Gate;I would cut my legs and tits off When I think of Boris Karloff an
M;Creeping Death;Slaves Hebrews born to serve, to the pharaoh Heed To his every
N;Been A Son;She should have stayed away from friends She should have had more
M;Cyanide;Sleep and dream of this Death angel’s kiss Brings final bliss Completely I
M;Better Than You;I look at you, then you me Hungry and thirsty are we Holding the
N;Love Buzz;Would you believe me when I tell you You are the queen of my heart P
M;Halo On Fire;Obey Obey Come won't you stay Sincere Sincere All ends in tears Er
N;Downer;Portray sincerity Act out of loyalty Defend every country Wish away the
M;Sweet Amber;[x2] Wash your back so you won't stab mine Get in bed with your o
```

length : 222.357   lines : 209   Ln : 26   Col : 29   Sel : 0 | 0   Windows (CR LF)   ANSI   INS

Figure 2. Dataset

The last step in dataset preprocessing were common transformations: changing all the letters into lowercase, removing the stop words, numbers, punctuation and white spaces, and finally, stemming. Sample of lyrics before and after transformations is shown on Figure 3.

And as the soft pretentious mountains Glisten in the light of the trees And the flowers sing in D minor And the birds fly happily We'll be together once soft pretenti mountain glisten light tree flower sing d minor bird fli happili togeth
I'm diggin' my way I'm diggin' my way to something I'm diggin' my way to somethin' better I'm pushin' to stay I'm pushin' to stay with something I'm pushin' to stay with something diggin way diggin way someth diggin way somethin better pushin stay pushin stay someth pushin stay someth
Tear it down Strip the layers off My turpentine Old paint, old looks Cover up the past White heat, white light Super white bones Bones of you and I Pure if I... can't you help me? tear strip layer turpentin old paint old look cover past white heat white light super white bone bone pure help

Figure 3. Lyrics before and after transformations

After preprocessing document-term matrix was created. Dictionary consists of 2.932 terms and the document-term matrix sparsity is 98%. List of the most frequent terms for both artists is shown in Table 2.

Table 2. Most frequent terms

Metallica		Nirvana	
term	**freq**	**term**	**freq**
see	157	like	85
never	156	yeah	67
just	152	know	57
now	141	take	48
one	130	got	47
come	128	feel	46
take	128	one	46
away	118	said	46
feel	118	away	45
life	110	can	44
will	106	way	41
time	96	never	36
let	95	make	35
death	87	get	34
way	87	love	34
can	86	want	33
die	85	just	32
like	85	think	32
want	82	mind	30
day	81	see	30

The threshold of frequent words was set to 8. Words that appeared less than 8 times were eliminated before training the model. The model showed the best results when threshold was 8 or 9. By increasing or decreasing the threshold, the classifier made more incorrect decisions.

B. *Results and evaluation*

Results of classification by Naive Bayes are shown in confusion matrix (Table 3). It can be noticed that the classifier incorrectly prognosed performer 8 out of 69 times.

Table 3. Confusion matrix

Predicted \ Actual	Metallica	Nirvana
Metallica	40	6
Nirvana	2	21

Cases of special interest are those in which classifier did not make the correct decision. Looking at probabilities in such cases (Table 4), it is noticeable that in five of the eight cases the classifier was very confident (more than 95 %) in his decision.

Table 4. Probabilities in incorrect decisions

Actual	Predicted	Probability Metallica	Probability Nirvana
Metallica	Nirvana	0.02909	0.97091
Nirvana	Metallica	0.97369	0.02631
Nirvana	Metallica	0.54434	0.45566
Nirvana	Metallica	0.99939	0.00061
Nirvana	Metallica	0.52747	0.47253
Metallica	Nirvana	0.02356	0.97644
Nirvana	Metallica	0.52038	0.47962
Nirvana	Metallica	0.99992	0.00008

For the evaluation measures computing, in the confusion matrix class of interest is Metallica and it represents a positive class. Therefore, Nirvana is a negative class.

The accuracy is (40 + 21) / (40 + 21 + 6 + 2) = 61/69 = 0.88406
The error is 1 – 0.88406 = 0.11594
The precision is 40 / (40 + 6) = 40/46 = 0.86957
The recall is 40 / (40 + 2) = 40/42 = 0.95238
The F-measure is (2 × 0.86957 × 0.95238) / (0.86957 + 0.95238) = 1.65632 / 1.82195 = 0.90909

ROC curve is shown on Figure 4. Area under the ROC curve is 0,969, which is a very good result.

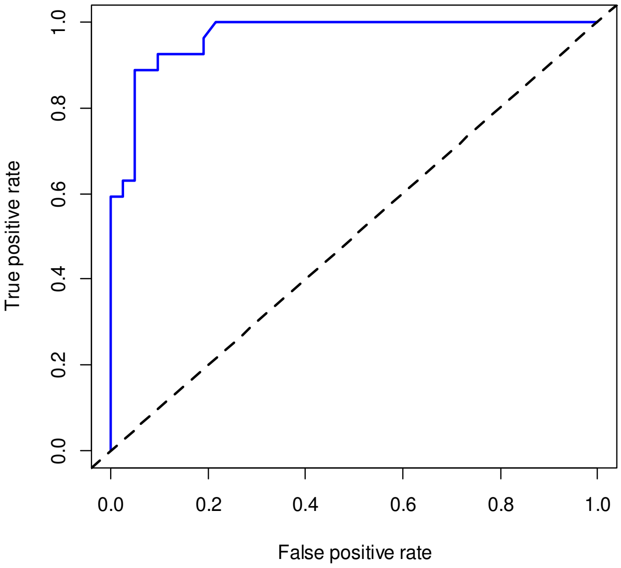


Figure 4. ROC curve

C. *Model improvement*

In order to improve the model, Laplace smoothing was applied. The best results model gave when Laplace estimator's value was 0.06. With this adjustment, classifier correctly recognized three more Nirvana's songs (Table 5).

Table 5. Confusion matrix of improved model

Actual Predicted	Metallica	Nirvana
Metallica	40	3
Nirvana	2	24

In the end, we checked the performance of the model without two transformations: removing the stop words and word stemming. The results without Laplace smoothing and with it were identical. Table 6 shows comparison of results.

Table 6. Comparison of results

Measure	Without Laplace smoothing	Laplace estimator = 0.06	Without stemming and removing stop words
precision	0.86957	0.93023	0.88636
recall	0.95238	0.95238	0.92857
F_1-measure	0.90909	0.94117	0.90697

IV. CONCLUSION

Creating a dataset was tedious and time-consuming task, partly because it was created manually, and partly because of doubt about inserting some songs into a dataset. Namely, cases such as guest appearances of other musicians on the album or two versions of the same song (a studio and a slightly altered live version) had to be handled with care. Besides, it was not always clear whether a song belong to a performer or not – the doubt was resolved with a help of Wikipedia's list of songs recorded by chosen artist.

Results of a created model are very good. Naive Bayes classifier is a good choice for this task – once again it proved its capabilities. Since the dataset was quite small, it was a logical candidate for the model.

Result showed that Nirvana's and Metallica's songs have textual 'signatures' that can be distinguished to a large degree solely on reading text. Results are more interesting when one takes into account the fact that songs for one band are often written by more authors. In some future research, it would be interesting to examine how the model behaves in a larger number of classes (artists) and to compare result obtained by Naive Bayes classifier with results obtained by other classifiers, especially with support vector machines.

LITERATURE

[1] A. Khan, B. Baharudin, L. H. Lee & K. Khan, "A review of machine learning algorithms for text-documents classification", Journal of advances in information technology, 1(1), 2010, pp. 4-20.

[2] X. Hu & J. S. Downie, "When Lyrics Outperform Audio for Music Mood Classification: A Feature Analysis", 11th International Society for Music Information Retrieval Conference (ISMIR), August 2010, pp. 619-624.

[3] M. Haggblade, Y. Hong & K. Kao, "Music genre classification", Department of Computer Science, 2011, http://cs229.stanford.edu/proj2011/HaggbladeHongKao-MusicGenreClassification.pdf .

[4] E. Stamatatos & G. Widmer, "Music performer recognition using an ensemble of simple classifiers", ECAI, 2002, pp. 335-339.

[5] M. Fell & C. Sporleder, "Lyrics-based analysis and classification of music", Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, 2014, pp. 620-631.

[6] X. Hu, J. S. Downie & A. F. Ehmann, "Lyric text mining in music mood classification", American music, 183.5, 049, 2009, 2-209.

[7] S. Howard, C. N. Silla Jr & C. G. Johnson, "Automatic lyrics-based music genre classification in a multilingual setting", Proceedings of the Thirteenth Brazilian Symposium on Computer Music, 2011.

[8] N. Homem & J. P. Carvalho, "Authorship identification and author fuzzy "fingerprints"", Fuzzy Information Processing Society (NAFIPS), 2011 Annual Meeting of the North American, IEEE, 2011, pp. 1-6.

[9] E. Stamatatos, "A survey of modern authorship attribution methods", Journal of the Association for Information Science and Technology 60.3, 2009, pp. 538-556.

[10] B. Lantz, "Machine learning with R", Packt Publishing Ltd, 2015.

[11] K. Ramasubramanian & A. Singh, "Machine Learning Using R", Apress, 2017.

[12] F. Peng, "Augmenting Naive Bayes Classifiers with Statistical Language Models", Computer Science Department Faculty Publication Series, Paper 91, 2003.

[13] C. Sammut & G. I. Webb, "Encyclopedia of machine learning and data mining", Springer, 2017.

[14] P. Cichosz, "Data mining algorithms: explained using R", John Wiley & Sons, 2015.

[15] D. M. Powers, "What the F-measure doesn't measure: Features, Flaws", Fallacies and Fixes. arXiv, 2015.