Application of the Naive Bayesian Classifier in Work on Sentimental Analysis of Medical Data

Nataliya Boyko [a], Karina Boksho [a]

[a] *Lviv Polytechnic National University, Profesorska Street 1, Lviv, 79013, Ukraine*

Abstract

This work includes study and analysis of the functional implementation and usefulness of the Naive Bayesian classifier, especially working with medical data. This article presents a model for the classification of controlled moods based on a naive Bayes algorithm. Naive Bayes is known to be one of the simplest probability classifiers. Typically, it works extremely well under favorable circumstances, despite the fact that all functions are conditionally independent of a specific class. In order to train such a classifier, it is important to measure the probabilities of classes as well as their conditional probabilities, which will later be used for new classifications.

Keywords 1

Naive Bayesian Classifier, Sentiment Analysis, KNN–k-nearest neighbor algorithm, Support Vector Machines

1. Introduction

Emotion recognition, in other words, the study of thoughts, is a large space for the study of judgments, beliefs, behaviors, as objects of the emotional fund for something particular. An entity, individual, product, or service, for example. At this stage, all of the above theoretical studies are under the aegis of mood analysis and thought extraction. If we single out the industry, this word can be found in a more scientific hue. The very analysis of sentence terms first appears in [1]. A substantial increase in text data with a bright saturated color that carries informative value involves an examination of the concept of mood expression and function focused, in particular, on the concept of business and its teachings.

SA 's application is to collect input from consumers on the introduction of new goods, political campaigns and even widespread in financial markets. The purpose of this strategy is to decide the attitude of the narrator to any subject or simply to the contextual polarity of the paper. Early work in this area was done by Terny and Peng ([2], [3]), who used various methods to determine the polarity of product and film reviews.

These days present day clinics are well prepared with observing and other information collection gadgets coming about in colossal information that are collected persistently through wellbeing examination and therapeutic treatment. All this is driven to the reality that the restorative zone produces progressively voluminous sums of electronic information which are getting to be more complicated.

Mood analysis is a challenging task, with the use of NB (Naive Bayes), K-NN (k-nearest neighbor algorithm) and SVM(Support Vector Machines) experiments.

The area of big data and machine learning may be the functional field of application of the findings of scientific work.

IDDM’2020: 3rd International Conference on Informatics & Data-Driven Medicine, November 19–21, 2020, Växjö, Sweden
EMAIL: nataliya.i.boyko@lpnu.ua (N. Boyko); boksho.karina@gmail.com (K. Boksho)
ORCID: 0000-0002-6962-9363 (N. Boyko); 0000-0001-7471-7328 (K. Boksho)

© 2020 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-2753/paper16.pdf

The goal of the work is to carry out a thorough analysis of the Naive Bayesian Classifier in comparison with some of the most common rivals of this technology in order to improve data processing. The proposed classification of the text, based on the collection of features and pre-processing, is therefore intended to serve as an opportunity to enhance the accuracy of the classification.

The key tasks in the course of the work are to establish the a priori concepts of the work of the Naive Bayesian classifier:

- describe the key characteristics, advantages and disadvantages of using NBC (Naive Bayesian Classifier) for sentimental analysis;
- define the key properties, advantages and disadvantages of using sentimental research help vector machines;
- define the key properties, advantages and disadvantages of using KNN-Method K of the closest neighbors for sentimental analysis;

The goal of the research is the problem of step-by-step data processing and the classification by vector of sentimental analysis of all the above methods and the analysis of the consequent optimization of its function. The topic of the research is the Naive Bayesian Classifier algorithm and its efficiency, which is compared with competitive means [7].

Acuteness of the study: analysis of attitudes is a method of collecting knowledge from the perceptions of users. People's decisions are affected by the views of others. Today, if someone wants to buy a product or wants to watch a movie, he/she will first look for feedback and opinions about that product or movie on social networks, blogs, etc. When there is a massive influx of user opinions on social networks such as Twitter, Facebook and other user forums, it becomes very difficult to classify moods with this large data manually. There is also a need for an integrated mood analysis framework.

In the job, various testing methods are used. Theoretical research methods include: algorithm analysis, comparison, convergence approach from abstract to concrete. Empirical approaches, including comparison and calculation, are directly present

2. Review of literature sources

The NBC is based on the Bayesian law, with a clear presumption of freedom. The naive Bayesian model presupposes a simplification of the conditional assumption of independence. In other words, a class (positive or negative) is given whose words are conditionally independent of each other. This assumption does not have a direct impact on the accuracy of the text classification, but actually allows the quick classification algorithms applicable to this mission. In their 2003 paper, Rennes et al. address the implementation of the Naïve Bayesian tasks of text classification. [6]

The main reason is that NB (Naïve Bayesian) with sampling tends to achieve a lower classification error than the original [6-9]. It has been shown that the performance of the NB classifier is significantly improved when sampling traits using an entropy-based method [12].

2.1. General representation of the Naive Bayesian algorithm

The NBC is based on the Bayesian law, with a clear presumption of freedom. The naive Bayesian model presupposes a simplification of the conditional assumption of independence. In other words, a class (positive or negative) is given whose words are conditionally independent of each other. This assumption does not have a direct impact on the accuracy of the text classification, but actually allows the quick classification algorithms applicable to this mission. In their 2003 paper, Rennes et al. address the implementation of the Naïve Bayesian tasks of text classification. [4]

NBC is a tool that applies to a particular class of tasks, namely those that are formulated to connect an object with a discreet category. From a community of numerical methods, the naive Bayes has a range of advantages, such as simplicity, speed and high precision. K. Ming Leung [5],[6] defines the law of Bayes.

2.2. General representation of the KNN algorithm

The k-nearest neighbor algorithm (k-NN) is a method of classifying an object based on the majority class among its nearest neighbors. KNN is a form of lazy learning in which the function is only approximated locally and all calculations are deferred to classification. The KNN algorithm is typically based on the Euclidean or Manhattan distance. However, you can use some other distance, such as the Chebyshev standard or the Mahalanobis distance. The major downside of KNN is that it uses all the functions to measure distance and costs a lot of time to identify objects [8].

2.3. General representation of the SVM algorithm

SVM works well for text classification because of its advantages, such as its ability for processing large items. Another benefit is that SVM is efficient when there are few instances, and also because most of the problems are linearly separated. The reference vector machine has shown promising results in previous studies in the field of mood analysis. [7] Reference vector machines are working on the concept of decision-making plans that establish decision-making boundaries. Many items belonging to various classes of association are divided into decision-making planes [10].

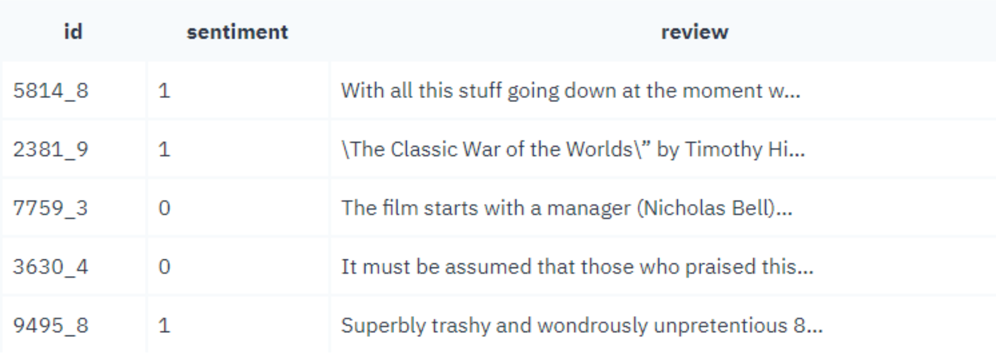
3. Primary processing

The accuracy of the results of the intellectual study is directly influenced by the quality of the data. The pre-processing step is therefore necessary in order to achieve a better classification result and even to improve the time used to train and generalize the model.

3.1. Dataset. Data description

The data comes from Kaggle's call - "Bag of Words Meets Bags of Popcorn". There are 25.000 IMDB movie reviews that are either positive or negative. IMDB scores are considered to range from 0 to 10. The additional pre-processing step performed by the data set authors transforms the rating into binary moods. Of course, one film can have several ratings, but with a condition of no more than 30.

The id column combines the movie ID with a unique number of reviews.



id	sentiment	review
5814_8	1	With all this stuff going down at the moment w...
2381_9	1	\The Classic War of the Worlds\" by Timothy Hi...
7759_3	0	The film starts with a manager (Nicholas Bell)...
3630_4	0	It must be assumed that those who praised this...
9495_8	1	Superbly trashy and wondrously unpretentious 8...

Figure 1: Example dataset

We're going to concentrate on the mood columns and reviews. The ID column is a combination of the movie ID and the unique review number. This may be relevant knowledge in real-world situations, but we're going to keep it easy.

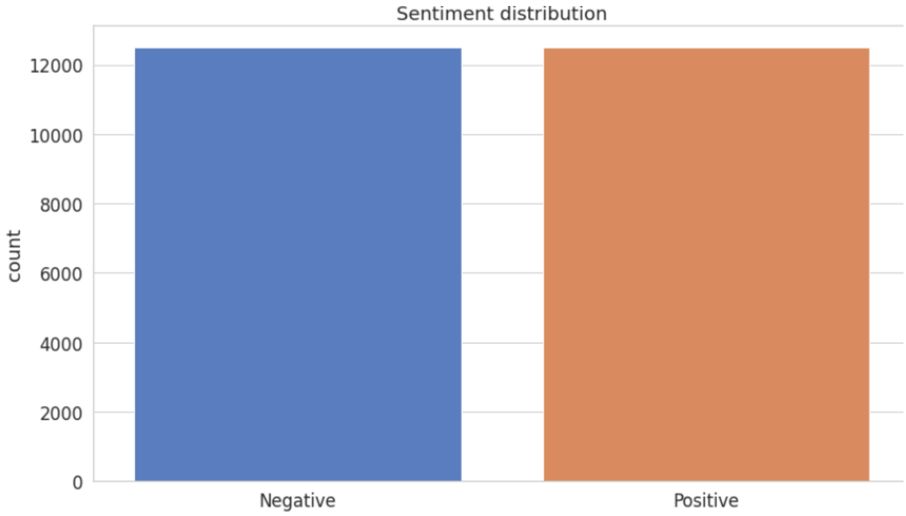
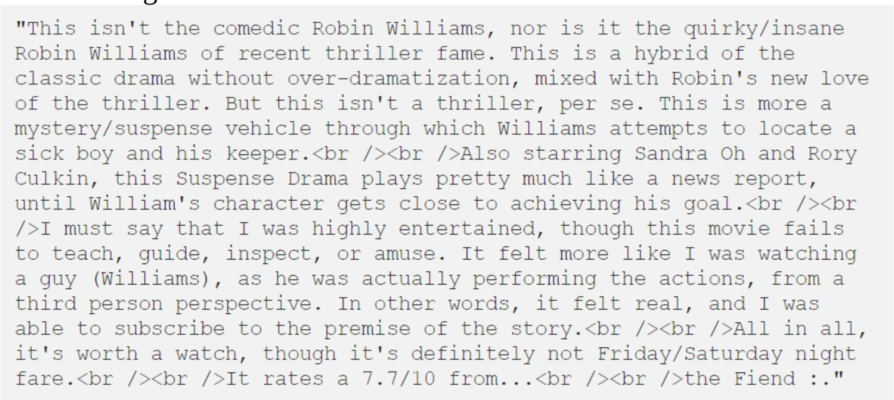


Figure 2: Representation of data distribution

Apparently, the existence of positive and negative behaviors is one-dimensional. The "raw" text is dirty enough for these reviews, so we need to clear everything before we can do some research. Here's an example of the following text:



```
"This isn't the comedic Robin Williams, nor is it the quirky/insane
Robin Williams of recent thriller fame. This is a hybrid of the
classic drama without over-dramatization, mixed with Robin's new love
of the thriller. But this isn't a thriller, per se. This is more a
mystery/suspense vehicle through which Williams attempts to locate a
sick boy and his keeper.<br /><br />Also starring Sandra Oh and Rory
Culkin, this Suspense Drama plays pretty much like a news report,
until William's character gets close to achieving his goal.<br /><br
/>I must say that I was highly entertained, though this movie fails
to teach, guide, inspect, or amuse. It felt more like I was watching
a guy (Williams), as he was actually performing the actions, from a
third person perspective. In other words, it felt real, and I was
able to subscribe to the premise of the story.<br /><br />All in all,
it's worth a watch, though it's definitely not Friday/Saturday night
fare.<br /><br />It rates a 7.7/10 from...<br /><br />the Fiend :."
```

Figure3: Type of text to be processed

In the analyses of these training sets, these are the most common words:

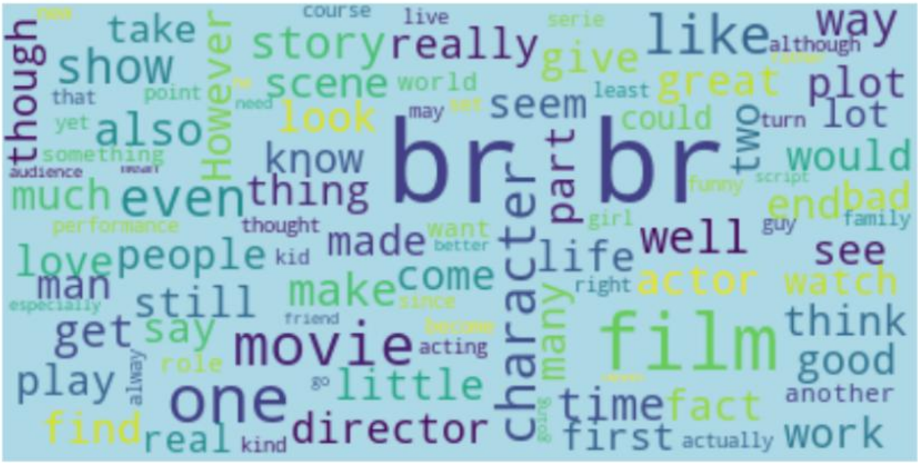


Figure 4: Representation of the most used words in the set

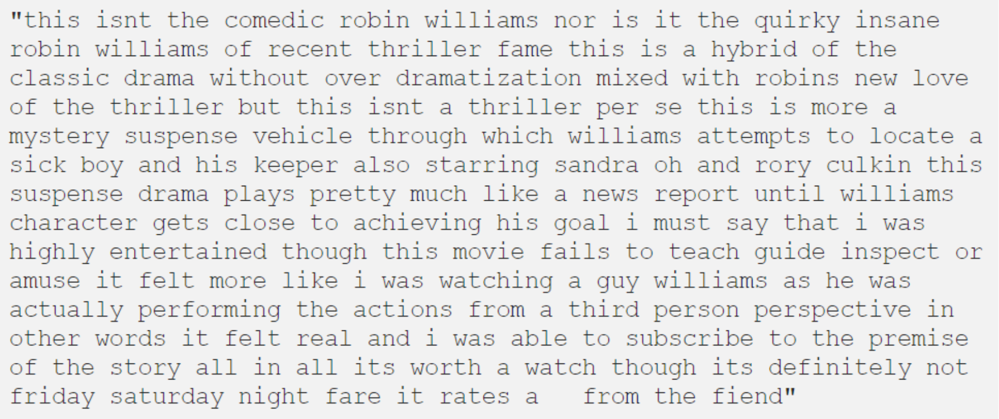
There is a question of a strange "br" in the set.

The so-called cleaning must be carried out. The chaos of data in the real world often crosses the line of absurdity [11]. They can, in turn, contain needless punctuation, HTML tags (as in the case of "br"), needless spaces, and so on.

We're doing a lot of cleanup with regular expressions, but we're going to use two libraries to handle HTML tags and delete common (stop) words.

We first use BeautifulSoup [8] to delete HTML tags from the text. In the future, remove all that is not a letter or space (including paying attention to ignoring capital letters) and replace the extra space with a single one.

Here is what the same text looks like in its purified form in the figure

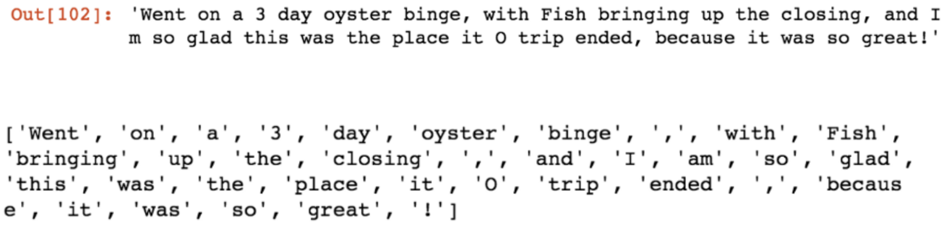


```
"this isnt the comedic robin williams nor is it the quirky insane
robin williams of recent thriller fame this is a hybrid of the
classic drama without over dramatization mixed with robins new love
of the thriller but this isnt a thriller per se this is more a
mystery suspense vehicle through which williams attempts to locate a
sick boy and his keeper also starring sandra oh and rory culkin this
suspense drama plays pretty much like a news report until williams
character gets close to achieving his goal i must say that i was
highly entertained though this movie fails to teach guide inspect or
amuse it felt more like i was watching a guy williams as he was
actually performing the actions from a third person perspective in
other words it felt real and i was able to subscribe to the premise
of the story all in all its worth a watch though its definitely not
friday saturday night fare it rates a   from the fiend"
```

Figure 5: Example of clean text

3.2. Tokenization

In this stage, the previous data is cleared, allowing you to continue the process and bring the data to the state of the Words Bag model [9]. By giving only lowercase letters to text data and splitting them into individual words, we apply the so-called tokenization. The last step in our pre-processing process is to delete stop words using those specified in the NLTK (Natural Language Toolkit) library [10, 14]. These are the ones that occur quite frequently, but do not bear any semantic loads. For instance, "a," "the," "and" Another reason for removing such stop words is [4, 15], without a doubt, the acceleration of execution, as we will certainly delete some of the results. Let's place our cleaning and tokenization function in a class called Tokenizer.



```
Out[102]: 'Went on a 3 day oyster binge, with Fish bringing up the closing, and I a
          m so glad this was the place it O trip ended, because it was so great!'


['Went', 'on', 'a', '3', 'day', 'oyster', 'binge', ',', 'with', 'Fish',
'bringing', 'up', 'the', 'closing', ',', 'and', 'I', 'am', 'so', 'glad',
'this', 'was', 'the', 'place', 'it', 'O', 'trip', 'ended', ',', 'becaus
e', 'it', 'was', 'so', 'great', '!']
```

Figure 6: Illustration of the text before and after the execution of the tokenizer

4. Method

Naive Bayesian models are probabilistic classifiers used by the well-known Bayesian theorem, perform and make clear assumptions about the independence of data features.

Intuitively, this may sound like a crazy idea. The following statement is a well-known fact: the previous word has an effect on the present and the next. The belief, however, simplifies mathematics and works very well in reality.

$$\gamma(\alpha,\beta)=\frac{\gamma(\alpha)\times\gamma(\alpha|\beta)}{\gamma(\beta)} \tag{1}$$

Where:

- specific class;
- a document that will be classified;
- a priori probability;

- a posteriori probability.

This equation gives us a conditional probability that event A will occur when B occurs. To find out, we need to measure the likelihood that B will happen if A occurs, and multiply it by the likelihood that A (known as the previous one) will happen. All of this is separated by the probability that B will happen on its own.

The naive assumption helps one to reformulate Bayes ' theorem as follows:

$$P(Sentiment|\omega_1,\ldots,\omega_n) = \frac{P(Sentiment)\prod_{i=1}^{n} P(\omega_i|Sentiment)}{P(\omega_1,\ldots,\omega_n)} \tag{2}$$

We just don't care about the odds. In a given case, we would like to know if the text has a positive or negative attitude. We can skip the denominator entirely, simply because it scales the numerator:

$$(Sentiment|\omega_1,\ldots,\omega_n) =\propto P(Sentiment)\prod_{i=1}^{n} P(\omega_i|Sentiment) \tag{3}$$

Thus, before choosing a feeling, we compare the scores for each feeling and select the one that has a higher score.

So, we're classifying the text into one of two groups / categories-positive and negative.

Multidimensional Naive Bayes helps us to present the features of the model in the form of the frequency of their occurrence (how much a term is present in our review). In other words, it informs us that the distributions of chance that we use are multinomial [13, 16].

The intuition of the classifier the text document is presented as if it were a bag of words, i.e. a split collection of words indicating their location with a bag of words, holding only their frequency in the document. In the example in the illustration, instead of reflecting the order of words in all phrases like "I love this movie" and "I would recommend it," we simply note that the word was repeated five times in the first passage, the word six times, the word love, recommend, film once, and so on.

The value of the α class can be positive / negative. A text is a summary of a particular film. The Naive Bayes Multinomial Model [11, 13] gathers knowledge on the frequency of words in documents. This approximation is shown in equation (4) for a priori probability.

Let us remember, first of all, the estimation of the highest probability. We're only going to use frequencies in the results. In the previous paper, we ask what percentage of the documents in our study set are in each class. Let the number of documents in our class C training details, and the total number of documents. And then:

$$\gamma(\alpha) = \frac{N_c}{N} \tag{4}$$

Where:

- the number of documents in the class;
- total number of documents.

Multinomial Naive Bayes helps you to present the characteristics of the models as the frequency of their occurrence.

The model is based on the multiplication of a number of probabilities. They can be so close that they're rounded to zero by the machine. We can therefore use the logarithmic probability:

$$\log P(Sentiment|\omega_1,\ldots,\omega_n) = \log P(Sentiment) + \log\prod_{i=1}^{n} P(\omega_i|Sentiment) \tag{5}$$

There is, however, an issue with learning with the highest probability. It suffices to say that we are attempting to make a positive evaluation of the likelihood of the word "fantastic" in this class, but suppose that there are no educational documents that simultaneously contain the word "fantastic" and are graded as positive. Perhaps the word "fantastic" appears by chance (in a sarcastic/ironical sense) in the negative class. In this case, the likelihood for this function is zero:

$$\hat{P}(fantastic\ |positive) = \frac{count("\,fantastic", positive)}{\sum_{\omega\in V} count(\omega, positive)} = 0 \tag{6}$$

But since naive Bayes naively multiplies all the probabilities of traits together, zero probability in terms of plausibility for any class will cause the probability of that class to be zero, regardless of other proof.

$$\hat{P}(\omega_i \ |c) = \frac{count(\omega_i, c) + 1}{\sum_{\omega \in V}(count(\omega, c) + 1)} = \frac{count(\omega_i, c) + 1}{(\sum_{\omega \in V} count(\omega, c) + |V|} \tag{7}$$

The problem with the MLE score is that it is zero for a term-class combination not contained in the training results. Training data are never large enough to accurately reflect the frequency of unusual occurrences. To remove the zero likelihood problem [12, 16], add-one or Laplace smoothing is used. This mainly adds one to each account. Add-one smoothing can be interpreted as a previous homogeneous one (each term occurs once for each class), which is then updated as the learning data is received. As a consequence, the probability of a document given by its class is the normal multinomial distribution previously presented in equation 2. Calculate the a priori probability of a positive negation using equation 5 as follows:

$$\gamma(pos) = \frac{3}{4}$$

$$\gamma(neg) = \frac{1}{4}$$

Let's calculate the maximum probabilistic smoothing of the Naive Bayesian estimate using equation 5:

$$\gamma(like|pos) = \frac{(3+1)}{25+31} = \frac{4}{56} = \frac{1}{14}$$

$$\gamma(boring|pos) = \frac{(0+1)}{25+31} = \frac{1}{56}$$

$$\gamma(good|pos) = \frac{(1+1)}{25+31} = \frac{2}{56} = \frac{1}{28}$$

Table 1

Calculate the a priori probability of a positive negation

data	Training				Test
doc	1	2	3	4	5
words	I like movie. It's lovely history.	Acting is Pretty well, I like it but heroin role is bad. Overall movie is marvelous.	I like picture, which is so melancholic.	Story is good but ending is so boring and sadly.	I like director's direction. The location place in movie is so boring. But story is good.
class	pos	pos	pos	neg	x

$$\gamma(like|neg) = \frac{(0+1)}{12+31} = \frac{1}{43}$$

$$\gamma(boring|neg) = \frac{(1+1)}{12+31} = \frac{2}{43}$$

$$\gamma(good|neg) = \frac{(1+1)}{12+31} = \frac{2}{43}$$

A posteriori probability is calculated:

$$\gamma(pos|doc5) = \frac{3}{4} \times \frac{1}{4} \times \frac{1}{56} \times \frac{1}{28} = 3{,}4165e^{-5}$$

$$\gamma(neg|doc5) = \frac{1}{43} \times \frac{2}{43} \times \frac{2}{43} = 1{,}2577e^{-5}$$

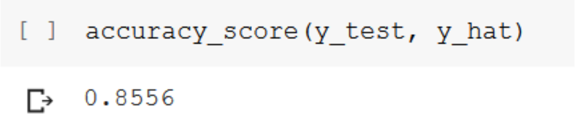
$$\gamma(pos|doc5) > \gamma(neg|doc5)$$

$\gamma(pos|doc5)$ – the maximum average probability of positive words in document 5 is maximum, so document 5 is positive.

5. Model evaluation

We have an average classification accuracy of 86 per cent in a collection of 25.000 film reviews. The basic algorithm is designed to train O (n + V lg V) and O (n) for testing, where n is the number of words in the document (linear) and V is the size of the abbreviated dictionary. This is much faster than other machine learning algorithms, such as Maxent classification or support vector machines, which take a long time to get close to the optimal weight range. This accuracy is comparable to the accuracy of current algorithms used to identify moods in film reviews [13, 15].

So, you should start by defining a number of variables and grouping the data by class. As we can see from the performance, the best accuracy of ~86% was achieved on the test set.



```
[ ]  accuracy_score(y_test, y_hat)

     0.8556
```

Figure 7: Representation of the best achieved accuracy

Classification report:

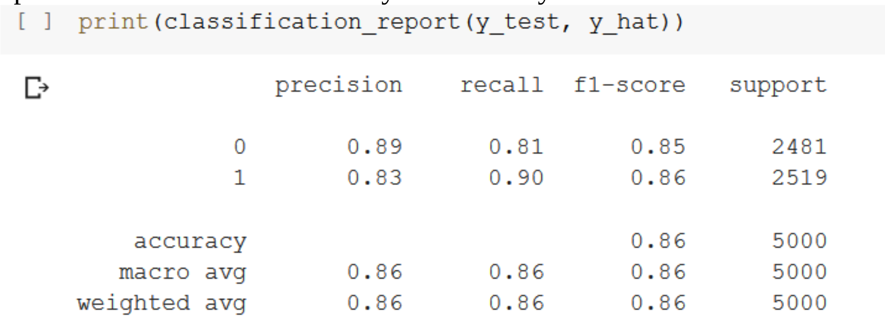
By comparing a valid instance class that was previously generated by the classification model, the performance of such a system will be measured in terms of recall, accuracy, and F-measure. For their mathematical definition, the following formulas will serve:

$$recall = \frac{Number \quad of \quad documents \quad retrieved \quad that \quad are \quad relevant}{Total \quad namber \quad of \quad documents \quad that \quad are \quad relevant} \qquad (8)$$

$$presicion = \frac{Number \quad of \quad documents \quad retrieved \quad that \quad are \quad relevant}{Total \quad namber \quad of \quad documents \quad that \quad are \quad relevant} \qquad (9)$$

$$F - measure = \frac{2 \times recall \times presicion}{recall + presicion} \qquad (10)$$

Now let's look at other metrics such as accuracy, recall, and F1 score (formula 10) to take a critical look at the situation. This is why, instead of being reliable on its own, we generally refer to two other metrics: precision and recall and F1. Precision tests the percentage of the elements detected by the system (i.e. the system is labeled positive) that are actually positive (i.e. positive according to their gold labels). Precision is defined as (9) Recall measures the percentage of elements currently present in input data that have been correctly identified by the device. The callback is described as (8).



```
[ ]  print(classification_report(y_test, y_hat))

                   precision    recall  f1-score   support

              0         0.89      0.81      0.85      2481
              1         0.83      0.90      0.86      2519

       accuracy                             0.86      5000
      macro avg         0.86      0.86      0.86      5000
   weighted avg         0.86      0.86      0.86      5000
```

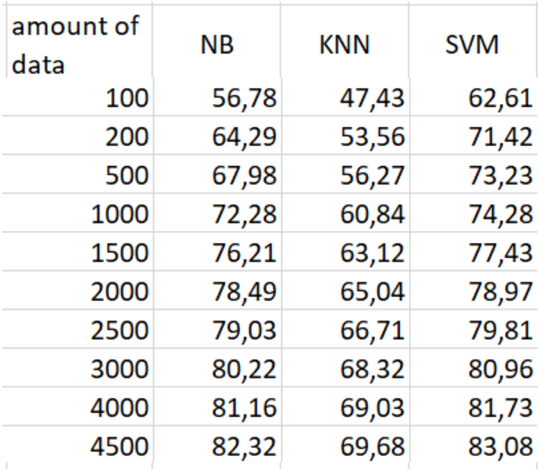
Figure 8: Estimated indicators

You can immediately see that Precision tells you how accurate/inaccurate your model is from those predicted positives, how many of them are actual positives. In our model 0.86.

In order to further assess the efficiency of the proposed pre-treatment stage, the outcomes of the previous and subsequent treatments are compared. However, if the results are worse than in the absence of a pre-processing period, which means that the classification model is not good enough, then changes are needed and the model is likely to be reconstructed. In addition, the naive classifier of Bayes will be checked with other classifiers (such as SVM, KNN) to show the superiority or refutation of the following: naive Bayes is better or at the same stage.

An F1 score is required when you need to find a balance between accuracy and recall. We have already shown that accuracy may be mainly attributed to a large number of real negatives, which, in most business situations, we do not concentrate on, though false-negative and positive generally have business costs (tangible and intangible), so the F1 score might be the best metric to be used if we need to find a balance between accuracy and response and the unequal distribution of classes (i.e. In this case, F1=0.86).

6. Evaluation of the effectiveness



amount of data	NB	KNN	SVM
100	56,78	47,43	62,61
200	64,29	53,56	71,42
500	67,98	56,27	73,23
1000	72,28	60,84	74,28
1500	76,21	63,12	77,43
2000	78,49	65,04	78,97
2500	79,03	66,71	79,81
3000	80,22	68,32	80,96
4000	81,16	69,03	81,73
4500	82,32	69,68	83,08

Figure 9: Comparison of accuracy on test data sets

Where NB is the naive Bayesian classifier, KNN is the nearest neighbor Method, and SVM is the support vector method.

Comparison of accuracy on test data sets (graphically):

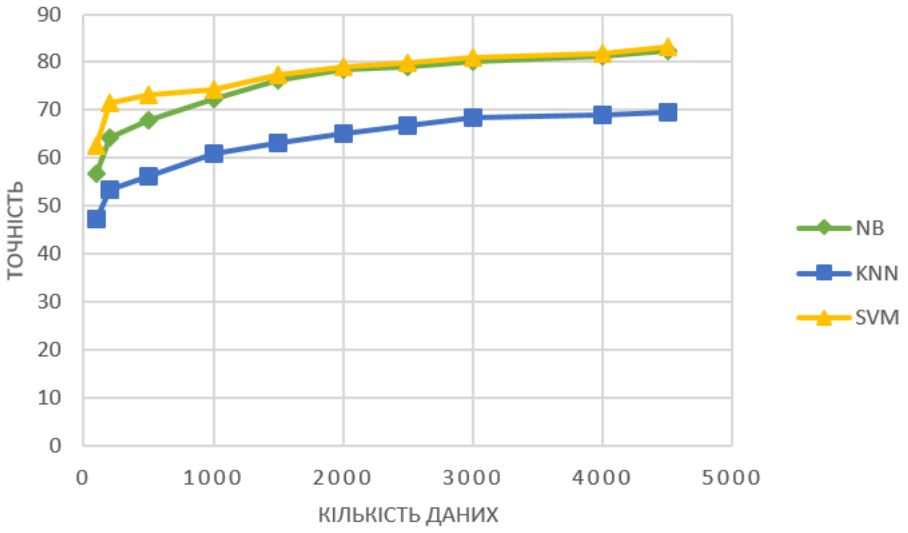


Figure 10: The interdependence of the accuracy of the data from quantity of data

With regard to the sophistication of the volume of data and the precision of the data, the methods under review have performed very well. As you can see, knn has shown the worst results, and naive

Bayes and SVM have been similar to each other, but the SVM approach has remained the leader in all measures of quantitative accuracy with qualitative indicators.

Method remained the leader in all indicators of objective accuracy with qualitative indicators.

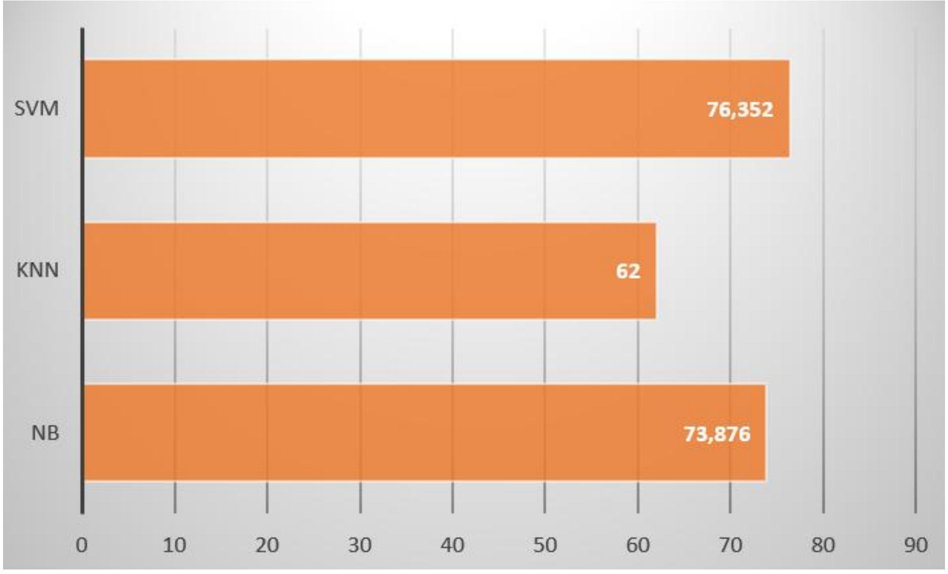


Figure 11: Comparison of average accuracy based on movie reviews

In all of the above statistics, a range of classifications are based on the accuracy of the findings. The naïve Bayes algorithm produces more reliable samples than the KNN algorithm, and the SVM produces more than the naive Bayes algorithm. The General SVM classifier therefore produces better results than the naive Bayesian and KNN classifiers.

7. Conclusion

The results show that a simple naive Bayesian classifier can be enhanced to match the accuracy of the classification of more complex mood analysis models by selecting the correct type of features and removing noise by selecting the wrong features.

Among the distinctive approaches, we have a tendency to emphasize the utility of Naïve Bayes (NB) that is one of the foremost compelling and effective classification calculations and has been effectively connected to numerous restorative issues.

Naive Bayesian Classifier:

- The naive classifiers of Bayes are linear classifiers based on the Bayes theorem. The resultant model is probabilistic.
- This is called naive on the basis that the objects in the data set are mutually independent.
- In the real world, independence assumptions are frequently broken, but naive Bayesian classifiers still appear to function very well.
- The aim is to break down all available data as predictors into a Bayes rule in order to provide a more reliable probability of predicting a class. It calculates the conditional likelihood, which is the likelihood that something is going to happen because something else has already occurred. For example, this review is likely to be negative, based on the existence of words like "bad"
- Relatively effective, simple to implement, fast and accurate, naive Bayesian classifiers are used in several different fields, as shown by analyzes of previous Hickey field studies.

1. Within the course of the work carried out, consideration was drawn to a few such points of interest: the convenience of usage is regularly the key advantage of Naïve Bayes. They were not able to be less precise than their a lot of complex partners, such as support vector machines and calculated relapse, but a few consider have appeared that considerably higher exactness can be accomplished.
2. Simple to apply.

Some of the weaknesses have been identified:

The point of zero frequency is well known. You may use the anti-aliasing technique to solve this problem. One of the simplest smoothing techniques is the calculation of Laplace.

The presumption of independent predictors is another weakness of naive Bayes. In real life, it's almost difficult to get a set of predictors that are totally independent.

Thus, considering its unrealistic presumption of independence, the naive Bayesian classifier is surprisingly successful in practice, since its classification solution can often be right and its probability estimates accurate. As we have shown, even a very simple implementation of the naive Bayes algorithm can result in surprisingly good results for sentiment analysis. Notice that this model is basically a binary classifier, which means that it can be used for any dataset that has two categories. There are all sorts of applications for this, from spam detection to sentiment-based bitcoin trading.

The study shows that the SVM Classifier performs a better analysis of the accuracy of the above data sets compared to the commonly used KNN and Naive Bayes machine learning classifiers.

Both of the above analyzes help us foresee the arrival of goods on the market that could boost the income of the crushed organizations.

8. References

[1] B. Pang, L. Lee, S. Vaithyanathan “Thumbs up? : sentiment classification using machine learning techniques”, In: Proceedings of the ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10. Association for Computational Linguistics, 2002, pp. 321-342.

[2] A.L. Maas, R.E. Daly, P.T. Pham, D. Huang, A.Y. Ng, C. Potts “Learning Word Vectors for Sentiment Analysis”, In: The 49th Annual Meeting of the Association for Computational Linguistics, ACL 2011, 2011, pp. 23-36.

[3] J.D. Rennie “Tackling the poor assumptions of naive bayes text classifiers”, In: Machine Learning-International Workshop then Conference, vol. 20(2), 2003, pp. 56-62.

[4] C. Tseng, N. Patel, H. Paranjape, T. Y. Lin, S. Teoh “Classifying twitter data with naive bayes classifier” in IEEE International Conference on Granular Computing, 2012, pp. 89-101.

[5] B. Pang, L. Lee, and S. Vaithyanathan “Thumbs up?: Sentiment classification using machine learning techniques,” in Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10, ser. EMNLP ’02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp.79-86.

[6] P. Vitynskyi, R. Tkachenko, I. Izonin and H. Kutucu, "Hybridization of the SGTM Neural-Like Structure Through Inputs Polynomial Extension," 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, 2018, pp. 386-391, doi: 10.1109/DSMP.2018.8478456.

[7] N. Boyko , L. Mochurad, I. Andrusiak, Yu. Drevnytskyi “Organizational and Legal Aspects of Managing the Process of Recognition of Objects in the Image”, Proceedings of the International Workshop on Cyber Hygiene (CybHyg-2019) co-located with 1st International Conference on Cyber Hygiene and Conflict Management in Global Information Networks (CyberConf 2019), Kyiv, Ukraine, November 30, 2019, pp. 571-592.

[8] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan, “Automatic sub-space clustering of high dimensional data”, vol. 11(1), Data mining knowledge discovery, 2005, pp. 5–33.

[9] V. Estivill-Castro, I. Lee, “Amoeba: Hierarchical clustering based on spatial proximity using Delaunay diagram” [9th Intern. Symp. on spatial data handling, Beijing, China, 2000, pp. 26–41].

[10] N. Boyko, N. Shakhovska “ Prospects for Using Cloud Data Warehouses in Information Systems”, 2018 in IEEE 13th International scientific and technical conference on computer sciences and information technologies (CSIT), vol. 2, DOI: 10.1109/STC-CSIT.2018.8526745

[11] D. Guo, D.J. Peuquet, M. Gahegan, “ICEAGE: Interactive clustering and exploration of large and high-dimensional geodata”, vol. 3, N. 7, Geoinfor-matica, 2003, pp. 229–253.

[12] D. Harel, Y. Koren, Clustering spatial data using random walks, Proc. of the 7th ACM SIGKDD Intern. conf. on knowledge discovery and data mining, San Francisco, California, 200, pp. 281–286.

[13] N. Boyko, O. Pylypiv, Yu. Peleshchak, Yu. Kryvenchuk, J. Campos "Automated Document Analysis for Quick Personal Health Record Creation" The 2 nd International Workshop on Informatics & Data-Driven Medicine (IDDM 2019), Volume 1. Lviv, Ukraine, November 11-13, 2019, pp. 208-221.

[14] C. Zhang, Y. Murayama, "Testing local spatial autocorrelation using", vol. 14, Intern. J. of Geogr. Inform. Science, 2000, pp. 681–692.

[15] N. Melnykova, V. Melnykov, E. Vasilevskis ”The personalized approach to the processing and analysis of patients' medical data". CEUR Workshop Proceedings, 2018, Vol. 2255: Proceedings of the 1st International workshop on informatics & Data-driven medicine (IDDM 2018) Lviv, Ukraine, November 28–30, 2018., pp. 103-112.

[16] V. Yakovyna, A. Peleshchyshyn, S. Albota ”Discussions of wikipedia talk pages: Manipulations detected by lingual-psychological analysis", CEUR Workshop Proceedings, 2019, Vol. 2392, pp. 309-320.