Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie

AGH University of Krakow

Sentiment analysis of movie reviews using Naive Bayes classification

Łukasz Obrzut Mateusz Mglej

AGH

Part I

Data collection

www.agh.edu.pl

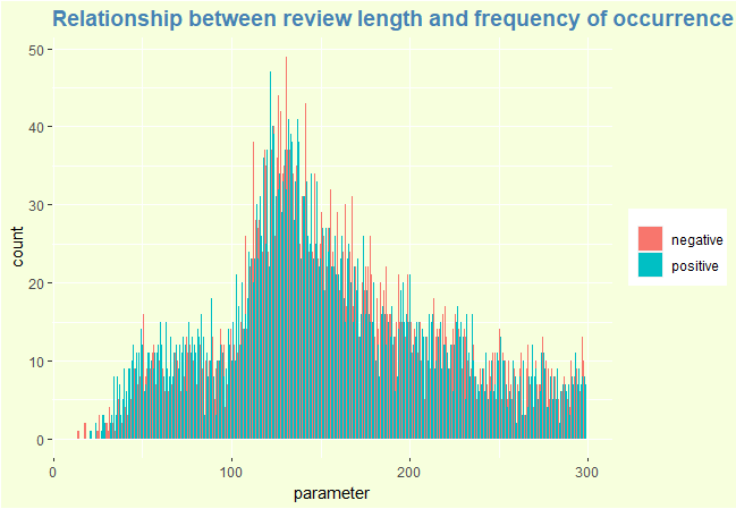
Database and problem statement

Our goal is to solve the problem of classifying movie reviews. For this project, we will use a dataset consisting of 50,000 movie reviews from IMDB, both positive and negative. We will build our model using a subset of 10,000 reviews (this will be justified later). Based on the number of reviews, word frequencies, and the appropriate Laplace smoothing parameter, we will apply suitable machine learning techniques for this type of task.

The database is available at:https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews.

Distribution of review lengths

The distribution of review lengths is similar for both positive and negative reviews.



Review examples

The data contains certain shortcomings (duplicates, HTML characters) and patterns (e.g. keywords for a given opinion), which we will take into account in our analysis.

Review examples

- positive: *"Clearly an hilarious movie.< br/ >< br/ >It angers me to see the poor ratings given to this piece of comic genius< br/ >< br/ >Please look at this for what it is, a funny, ridiculous enjoyable film. Laugh for christ sake!< br/ >< br/ >"*
- negative: *"This is the weakest of the series, not much of a plot and a rather odd-looking Wallace. But it's still pretty good, considering. A sign of greater things to come!< br/ >< br/ >6/10"*

AGH

Part II

Exploration and preparation of data

Clearing reviews

Data cleaning in our analysis will sequentially involve:

1. Removing duplicate reviews (418 cases),
2. Converting to lowercase,
3. Removing numbers and stop-words (common words with little significance for classification problems),
4. Removing HTML tags,
5. Removing punctuation marks,
6. Reducing words to their stems.

Comparing the original review to the cleaned one

AGH

Original review

"We brought this film as a joke for a friend, and could of been our worst joke to play. The film is barely watchable, and the acting is dire. The worst child actor ever used and Hasslehoff giving a substandard performance. The plot is disgraceful and at points we was so bored we was wondering what the hell was going on. It tries to be gruesome in places but is just laughable.$< br/ >< br/ >$Just terrible"

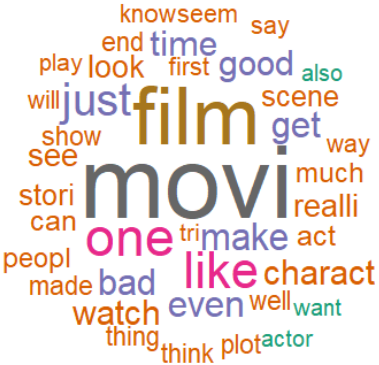
Cleaned review

"brought film joke friend worst joke play film bare watchabl act dire worst child actor ever use hasslehoff give substandard perform plot disgrac point bore wonder hell go tri gruesom place just laughabl just terribl"

Word cloud

From the cleaned data, we can graphically present the most frequently occurring words in our reviews using a word cloud.

Rysunek: Positive reviews



Rysunek: Negative reviews

Preparing Data for Classifier Construction

After cleaning the data, we can proceed to the following operations:

- Tokenization of the text, which involves splitting the text into words,
- Creation of a Document-Term Matrix (DTM), where each cell contains the number of occurrences of a given word in a review,
- Randomly splitting our data into training and test sets in a $3:1$ ratio,
- Removing potentially non-useful words, i.e., those that appeared in fewer than 10 reviews from the training set (Freq parameter $= 10$),
- Introducing indicator variables that will indicate whether a given word appears in a particular review or not.

AGH

Part III

Model construction

www.agh.edu.pl

The choice of data analysis tools

The Naive Bayes classifier may be suitable for our analysis because:

- We need to extract a subtle influence from a large number of features,
- It has low complexity (important since the Document-Term Matrix (DTM) is large),
- It is known for performing well with text classification tasks.

Specification of the Analytical Model Used

In our classification problem, the Bayesian probability will be equivalent to, for example, the problem

$$P(\textbf{negative}|\textbf{bad}) = \frac{P(\textbf{bad}|\textbf{negative})P(\textbf{negative})}{P(\textbf{bad})},$$

where:

- **negative** - the given review is negative,
- **bad** - the given review contains the word "bad".

In general, our model can estimate the probabilities of a review belonging to a class (positive or negative) given the occurrence of specific words $S_1, \ldots, S_n$ using the formula:

$$P(\text{review}_i|S_1, \ldots, S_n) = \frac{1}{Z}P(\text{review}_i)\prod_{i=1}^{n} P(S_i|\text{review}_i),$$

where Z is an appropriate scaling factor.

AGH

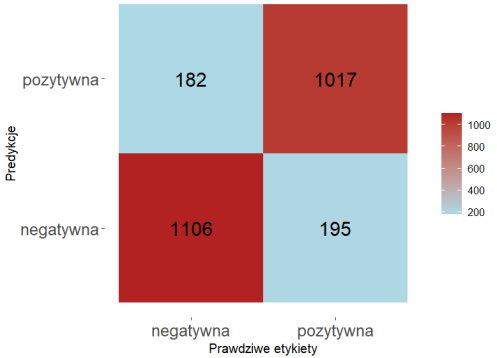
Part IV

Model evaluation

www.agh.edu.pl

Results

After training the classifier and evaluating the model's performance on the test set, we obtain an accuracy of **84.9%**.
Below, we present the results in the form of a cross table:



Rysunek: Prediction results for 2500 reviews

Model Evaluation

Results for the test set containing 2500 reviews:

- 2123 reviews correctly predicted,
- 377 errors.

Considering the simplicity of the algorithm and the complexity of the data (since sentiment sometimes depends on the context of the entire statement rather than individual words), the obtained accuracy of **84.9%** seems very reasonable.

What's next?

- Attempt to modify the parameters of our model,
- Apply a different classification model: decision tree.

AGH

Part V

Model Refinement

www.agh.edu.pl

Frequency Parameter

In the original model, the Frequency parameter (Freq) is set to 10, meaning the model ignores words that appear in fewer than 10 reviews from the training set.

Frequency Parameter (Freq)	Number of Words in Training Set	Accuracy
10	7171	84.9 %
5	11110	85.4 %
15	5533	85.0 %
20	4516	85.1 %
25	3939	84.9 %
30	3472	84.8 %

There is no significant improvement. The best results are for parameter values of 5 and 20. For practical reasons (computation speed), further refinements will use the parameter $Freq = 20$.

Laplace Parameter

Laplace smoothing helps avoid situations where the model incorrectly classifies a review because a keyword did not appear in the training reviews. By default, the model does not use this parameter, but using it may help refine the model.

Laplace Parameter	Accuracy
0	85.1%
0.5	85.1%
1	85.2%
5	85.1%
10	84.9%

There is no significant improvement; the best result is for the parameter value of 1.

Number of Data Used

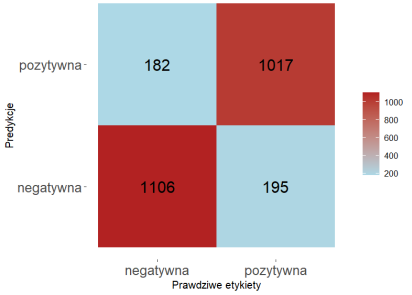
The initial model uses 10,000 reviews. For this value, the Frequency parameter is set to 20. Does increasing the number of data improve the model?

Number of Data	Frequency Parameter (Freq)	Accuracy
10,000	20	85.1 %
20,000	40	85.7 %
49,582	100	84.2 %

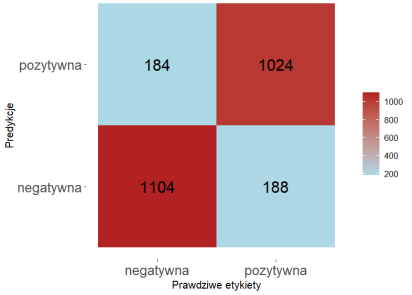
Using more data does not significantly improve the model. The better result at 20,000 is likely due to better data quality. It can be assumed that using 10,000 reviews is sufficient.

Summary of Parameter Adjustments

- Initial model: 10,000 data, Freq: 10, Laplace: 0 $\Rightarrow$ 84.9%.
- Refined model: 10,000 data, Freq: 20, Laplace: 1 $\Rightarrow$ 85.2%.



Rysunek: Initial model



Rysunek: Refined model

Decision Trees

Decision tree model using the C5.0 algorithm

Accuracy	Model Parameters
78.9%	no boosting
81.6%	trials: 8
81.7%	trials: 10
82.2%	trials: 13
82.3%	trials: 14
82.1%	trials: 15

The accuracy of the initial model is 78.9%. Attempting to improve it using the AdaBoost method (iteratively building multiple trees, parameter trials) yields better results but does not surpass the accuracy achieved with the naive Bayes classification.