Sentiment Analysis with Machine Learning on Reviews of Amazon Products

Miha Petrišič
Faculty of Computer and
Information Science
University of Ljubljana
1000 Ljubljana
Email: mp6079@student.uni-lj.si

***Abstract*—As a part of the research on sentiment analysis we have developed a number of different machine learning models to help us with extracting sentiment from text. We used Amazon's reviews on cellphone and accessories with a goal of predicting costumer's score from 1 to 5 based on their written review. We used natural language processing techniques like stop word removal, word correction and lemmatization on a subsection of 10.000 reviews. Before training the data we used document frequency as our feature selector. Based on our evaluations of the models support-vecrtor machine, Naive Bayes and multi-layer perceptron all performed very similarly with the classification accuracy of around 43-44% and mean absolute error of about 0,85. In general Naive Bayes performed best if we are looking for an efficient algorithm that brings satisfying results in a short amount of time.**

***Index Terms*—Opinion mining, Amazon products, feature selection, Support-vector machine, Naive Bayes, Multi-layer perceptron.**

I. Introduction

Sentiment analysis (SA) refers to a process used for extracting person's emotions or mood within a text. To achieve this it uses different techniques of natural language processing (NLP), computational linguistics and text analysis. The rise in SA occurred with the internet and the sheer availability of subjective texts on the web. Most of the research papers on the topic focused on analyzing product reviews while more recent ones focuses on SA based on social media texts.

Our research focuses on analysing reviews on cellphones and accessories submitted to Amazon's website. Our main goal consists of training the machine learning (ML) model so it can correctly predict the score that user gave on a certain product based on the same user's review. Main issues we tackled during the research were based on pre-processing the data in the way that is time efficient and gives satisfactory results. This includes scaling down the data, using different NLP techniques and selecting the right feature selector for the model to train on.

The following research studies related work mainly to help the reader to understand the standard practices used in SA. Related work also includes all important concepts that need to be understood in order to understand our methodology of work. We also present the results of our experimental work on different models. In the end we conclude with a summation of our work and reflection on our problems during the research process and possible solutions and improvements.

January 4, 2021

II. Related Work

Sentiment analysis (SA), also sometimes called opinion mining is a combination of different methods which aim to recognize the subjective information from text. With increasing popularity of data science and the usage of the worldwide web, computational intelligence has become an important tool for many companies. Since opinions are one of the biggest influences on human behaviour it has become an important task to analyze those opinions. Knowing your costumers is essential when designing new products. With SA and lots of available data in form of the reviews, it has become easier to extract costumers opinions on certain products. More satisfied costumers generate more revenue hence the understanding of costumers' perception is the key to success [1].

The process of SA usually involves 5 different steps [2]:

- data collection - this step includes extracting text data from various sites, such as blogs, product review and social media sites,
- text preparation - cleaning the data so that non-textual content is removed and that the texts have unitary form,
- sentiment detection - differentiating between objective and subjective based texts,
- sentiment classification - sentences are classified (positive/negative, good/bad),
- presentation of output - evaluating and visualizing our predictions.

There are two main sentiment classification approaches: ML and lexicon-based approach.

A. Lexicon-based Sentiment Analysis

Lexicon-based approach focuses on calculating the orientation of the text based on the semantic orientation of words or phrases in the text [3]. The basic assumption of this approach is that sentiment is related to the presence of certain words or phrases in the text. It uses a sentiment lexicon to determine the polarity of the text. It is used instead of the ML approach since it's fast and it's very easy for a human to understand [4].

Lexicon-based methods, although in general very time-efficient, can be inaccurate in the cases of more sophisticated and complex opinion texts. That makes them useful for analyzing simple text while more domain and context specific texts might be difficult to classify [5]. Because of the relatively good results and short run time, new techniques of combining ML methods with lexicon-based approach has emerged. The hybrid method tries to provide a trade-off between run time and accuracy.

B. Machine Learning Sentiment Analysis

ML is the other main approach to SA. Even though this approach is more complex when it comes to the preparation and the computation of data, it can produce satisfying results with more contextual texts than the lexicon-based method.

Throughout the years many different ML classifiers have emerged to help with SA. Most popular include: Support Vector Machine (SVM), Naive Bayes (NB), Decision Trees, Maximum Entropy and K-nearest neighbour (KNN).

In practice, before the actual implementation of the model, the pre-processing of the data is done. That includes applying some basic natural language processing methods like stop word removal, tokenization and stemming. After that the tokens are arranged according to their occurrence and frequency.

Next the feature selection method is selected based on the extracted tokens. Only the top n-ranked attributes are used for the actual training of the classifiers. The number of selected features (n) vary from small to large. Selecting the right feature selection method is important because it can reduce the original data set by removing irrelevant features which can in the end result in better model accuracy and faster running time of the algorithm.

Some of the feature selection methods include:

- document frequency,
- information gain,
- gain ratio,
- CHI statistic and
- relief-F algorithm.

out of these, document frequency is the simplest. This method defines an interval and removes every feature that is outside of the interval. The assumption of the method is, that very common and very rare features do not contribute to the improvement of the classification accuracy. The method was found to be preforming similar in some studies to information gain and CHI statistic method while also having a lower computational cost than the mentioned methods [6], [7].

C. Naive Bayes Classifier

One of the simpler methods of ML is NB classifier. It's assumption is that there is no link between words. The model is trained on usually two-class data (positive or negative sentiment). By counting how many time each word in a document occurs given the positive or negative sentiment of the text we can calculate the probability of the word. By the product of every word in the text we can then calculate the sentiment of that text.

NB classifier was studied on many different data sets and showed a variety of different results. Study that used Facebook's user's comments showed approximately 71% classification accuracy on a two class problem [8]. There was also a study conducted on movie and hotel reviews. The accuracy of the model in this research was highly dependent on the data set. On movie reviews the accuracy was about 82% while the accuracy on hotel reviews was lower with 55% [9]. In another study the Multinomial NB algorithm performed with the accuracy of 86% using tweets as training and testing data [10].

D. Support Vector Machine Classifier

SVM solves classification and regression problems by constructing sets of hyperplanes which are used as separators in a multi-dimensional space. It is a commonly used classifier in SA since it performs well when dealing with a high amount of features [11].

A research in 2004 showed a SVM model which performed with an accuracy of 86.5% on an four n-fold cross validation experiments [12]. Another research used the algorithm on Pang [13] and Taboada corpuses [14]. The SVM models performed with the highest accuracy of 77,5% in one of the conducted experiments on the Taboada corpus and approximately 88% on the Pang corpus [15].

E. Multi-Layer Perceptron

Multi-Layer perceptron (MLP) is a feedforward artificial neural network. It maps the input data sets to the appropriate output sets. For training it uses a supervised learning method called backpropagation.

Many of the recent researches focus on applying the neural network models to SA problems. One of the recent studies used MLP to classify movie and product reviews as positive or negative. The accuracy of the model was found to be performing simmilar to the SVM model with 81% accuracy on the movie data and 79% accuracy on the product review data [16]. Another study analyzed different ML methods on Twitter and Tumblr comments that were made on various topics. The study used 3 different classes to classify text (positive, negative and neutral). On the topic of Rio Olympics MLP had an accuracy of 67% on Twitter comments which is worse than the NB and SVM classifier. On Tumbler data it outperformed NB with accuracy of 65%. Similarly, on the topic on US presidential elections it performed worse than NB and SVM on Twitter data with 58% accuracy and again performed better than NB and worse than SVM on Tumbler data with 70% accuracy [17].

III. METHOD

A. Data Collection

For the purposes of the research we used the data of reviews on cellphones and accessories on Amazon's products. The whole data set consists of almost 200 thousand different reviews. Every review contained user's written text and summary of the user's review as well as numeric score from 1 to 5.

The following list shows the example of how one of the rows looks:

- reviewText - "It's OK, too small for my hand.",
- summary - "Two Stars",
- overall - 2

B. Preprocessing the Data

Before training the model we had to do some preprocessing on the data. This step is the key not only because it can produce better results, but also because it can scale down the size of the problem significantly. Our goal was to filter the data in a way that every review text has a certain form. That means removing certain words, symbols and converting sentences to it's most basic form. That way there is more connectivity between different texts and makes it easier for our ML algorithm to extract the right emotion from the review.

We started with removing the punctuation and putting every character in lower case. We also removed words shorter than three characters before we continued to language processing. We used Natural Language Toolkit to remove stop words (is, at, the, etc.). We used then used spelling correction of the text before lemmatization.

The data set for training and testing the model was composed of 10.000 reviews, 2000 of every score from 1 to 5.

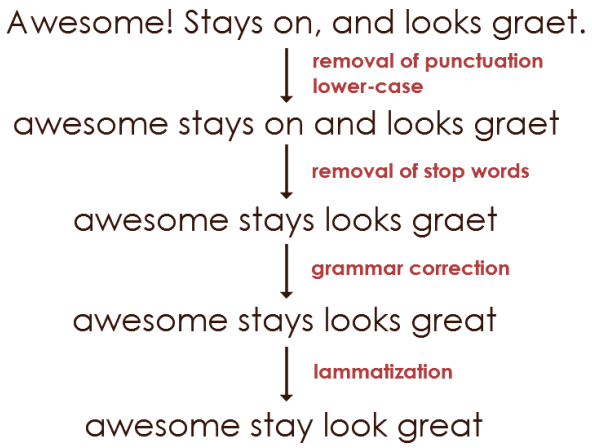

Fig. 1. Scheme of the text processing.

C. Feature Selection

Before training the model we had to transform the data in a way that model understands it. We created a feature extractor that builds a matrix based on words that appear in texts and the frequency of those words occurring in a review text. Because of memory and computational limitations we built a feature extractor that includes only words with a certain frequency. We used document frequency method as our feature selector which means that we cut of some of the features with lower frequency and some with highest frequency.

D. Training and Evaluating our Model

The data was trained and evaluated on the following models from Scilearn library:

- Naive Bayes classifier,
- Random forest classifier,
- k-nearest neighbours classifier,
- Support vector classifier and
- Multi-layer perceptron classifier.

We trained our models using different options and parameters. We also had to return back to feature designing with each model because of the different memory limitations that arose with different algorithms. To estimate the success of our model we used 10-fold cross-validation. With every iteration we computed classification accuracy (CA), mean squared error (MSE) and mean absolute error (MAE).

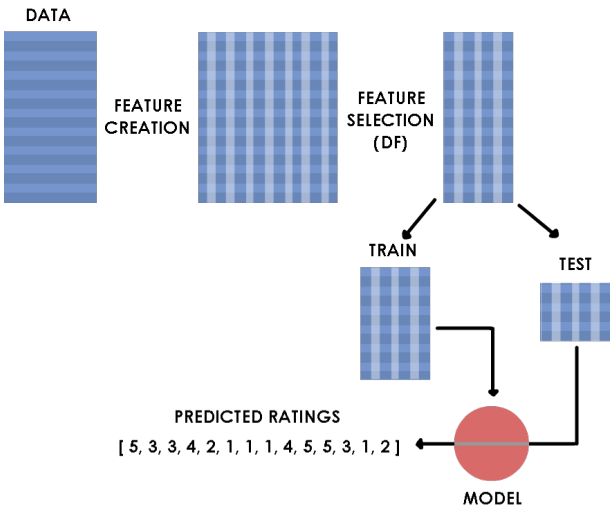

Fig. 2. Scheme of the total process.

IV. Results

One of the main goals of this research was to see how different methods of work produce different results. We were interested in the occurrence of certain words in reviews and how does the frequency of those words differ between reviews with different scores. We plotted word clouds for reviews with different scores to better visualize what kind of words appear in certain reviews.

Fig. 3. Word cloud of reviews rated with 5.

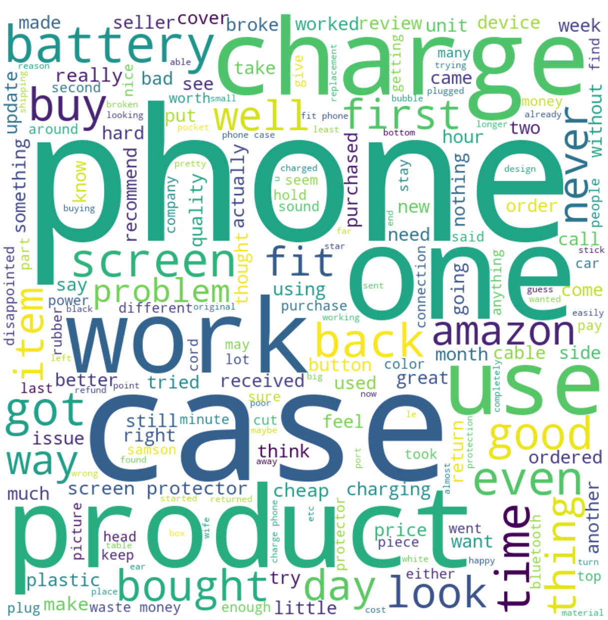

Fig. 4. Word cloud of reviews rated with 1.

The following tables present the performance of different ML algorithms using different parameters and feature selecting options.

A. *Naive Bayes*

TABLE I
RESULTS OF NAIVE BAYES CLASSIFICATOR

model	f	CA	MAE	MSE
Gaussian NB	0.10	0.31	1.30	3.15
	0.50	0.32	1.29	3.09
	0.95	0.32	1.29	3.08
Multinomial NB	0.10	0.43	0.87	1.69
	0.50	0.44	0.85	1.64
	0.95	0.44	0.85	1.64
Complement NB	0.10	0.41	0.94	1.95
	0.50	0.42	0.92	1.88
	0.95	0.42	0.92	1.88

f - feature extractor only takes f portion of the most frequent features.

B. *Random Forest*

TABLE II
RESULTS OF RANDOM FOREST CLASSIFICATOR

model	N	CA	MAE	MSE
Random forest	20	0.36	1.05	2.23
	100	0.41	0.94	1.94
	400	0.42	0.91	1.87

N - Number of trees in the forest.

C. *K-Nearest Neighbours*

TABLE III
RESULTS OF K-NEAREST NEIGHBOURS CLASSIFICATOR

model	K	CA	MAE	MSE
KNN	3	0.21	1.96	5.84
	5	0.22	1.85	5.37
	10	0.29	1.41	3.56
	20	0.34	1.17	2.66

K - number of neighbours.

D. *Support Vector Machine*

TABLE IV
RESULTS OF SUPPORT VECTOR MACHINE CLASSIFICATOR

model	strategy	CA	MAE	MSE
SVM	OVO	0.43	0.83	1.55
	OVR	0.41	1.89	1.72

OVO - "one-versus-one" approach for multi-class classification.
OVR - "one-versus-rest" approach for multi-class classification.

E. Multi-Layer Perceptron

TABLE V
RESULTS OF NEURAL NETWORK CLASSIFICATOR

model	algorithm	CA	MAE	MSE
Neural Networks	LBFGS	0.44	0.82	1.52
	ADAM	0.40	0.92	1.80

algorithm - the solver for the weight optimization.
LBFGS - optimizer from the family of quasi-Newton methods.
adam - stohastic gradient-based optimizer [18].

It is interesting to look at the specific reviews to see how the actual rating of the review differs from the predicted value. This can give us some insight on how our model works. It also shows what kind of sentences (like the second review in the following table) ML algorithm has problems with. The following table shows how the MLP model classified some of the reviews.

TABLE VI
PREDICTED RATINGS OF THE MLP ALGORITHM AND ACTUAL RATINGS ON A SPECIFIC REVIEW.

test review	actual rating	predicted rating
This item did not even work when I got it, even though I charged it for hours.....and hours. PASS on this item.	1	1
Car charger don't work, usb cord don't work as well. I didn't even try to plug in the wall charger. I am not a happy person right now.	1	5
to take the case off i had to take the pink part off then take the rubber off....please look for another case this is so not worth it.	3	2
So after many purchases and returns. This works well for us. My only one comment is I wish I could make the volume a bit louder.But even in the wind I am told people can hear me. I can hear on it well. Just with some minor hearing loss I would like it to be even louder. Holds a charge very well.	5	1

V. CONCLUSION

We researched the field of SA and used different models to extract emotional substance of the text. We used 5 different ML models out of which MLP, NB and SVM performed very similarly.

In general we are satisfied with our implementation of the ML algorithms. The models that performed well in previous researches worked similarly on our research of the problem. One of the issues with comparing our research to other ones done on SA is that we used a 5 class problems (ratings 1 - 5) while most other researches used only a 2 class classification (positive or negative). It would be interesting if we could convert our problem to a 2 class one, but then again, it would be difficult to determine the threshold for ratings to use to see weather a certain review is positive or negative.

Most of the issues arose with the memory limitations. When training and testing the models it was a constant problem of tweaking the feature selection algorithm to determine the threshold in a way that the program won't result in a memory error. Of course even the data set had to be filtered from the start because there was no way we could realistically process all of the 200 thousand reviews without running into memory issues.

As mentioned in the related work section there are many different feature selecting algorithm. One of the ides could be using a subset of them to try and test it on our problem. Based on the previous work done on feature selection it is very possible that some of them will improve the accuracy of our algorithm.

One of the possible improvements could also be a model that takes into an account the different combination of words as one feature. Word "good" might have a positive meaning, but when combining it with "not", the meaning changes dramatically.

Opinions can be expressed in many different ways. One of those is using emoticons. Further research should also put some attention to those representatives of emotions.

Besides ML there are also lexicon-based methods for SA. It is possible to combine both ML and sentiment lexicon in a hybrid method that might produce even better results.

REFERENCES

[1] M. Farhadloo and E. Rolland, *Fundamentals of Sentiment Analysis and Its Applications*, 03 2016, pp. 1–24.
[2] A. D'Andrea, F. Ferri, P. Grifoni, and T. Guzzo, "Approaches, tools and applications for sentiment analysis implementation," *International Journal of Computer Applications*, vol. 125, pp. 26–33, 09 2015.
[3] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 37, pp. 267–307, 06 2011.
[4] A. Jurek, M. Mulvenna, and Y. Bi, "Improved lexicon-based sentiment analysis for social media analytics," *Security Informatics*, vol. 4, pp. 1–13, 2015.
[5] Augustyniak, T. Kajdanowicz, W. Tuligłowicz, and P. Szymański, "Comprehensive study on lexicon-based ensemble classification sentiment analysis," *Entropy*, vol. 18, 12 2015.
[6] Y. Yang and J. Pedersen, "A comparative study on feature selection in text categorization," in *ICML*, 1997.

[7] S. Tan and J. Zhang, "An empirical study of sentiment analysis for chinese documents," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2622 – 2629, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417407001534

[8] C. Troussas, M. Virvou, K. Espinosa, K. Llaguno, and J. Caro, "Sentiment analysis of facebook statuses using naive bayes classifier for language learning," vol. 4, 07 2013, pp. 1–6.

[9] L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, "Sentiment analysis of review datasets using naïve bayes' and k-nn classifier," *International Journal of Information Engineering and Electronic Business*, vol. 8, pp. 54–62, 07 2016.

[10] R. Rajput and A. Solanki, "Real time sentiment analysis of tweets using machine learning and semantic analysis," 11 2016, pp. 687–692.

[11] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proceedings of the 10th European Conference on Machine Learning*, ser. ECML'98. Berlin, Heidelberg: Springer-Verlag, 1998, p. 137–142. [Online]. Available: https://doi.org/10.1007/BFb0026683

[12] T. Mullen and N. Collier, "Sentiment analysis using support vector machines with diverse information sources," in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 412–418. [Online]. Available: https://www.aclweb.org/anthology/W04-3253

[13] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," ser. ACL '04. USA: Association for Computational Linguistics, 2004, p. 271–es. [Online]. Available: https://doi.org/10.3115/1218955.1218990

[14] M. Taboada, C. Anthony, and K. Voll, "Methods for creating semantic orientation dictionaries," in *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. Genoa, Italy: European Language Resources Association (ELRA), May 2006.

[15] N. Zainuddin and A. Selamat, "Sentiment analysis using support vector machine," 09 2014, pp. 333–337.

[16] M. S. Husain and P. k. Singh, "Methodological study of opinion mining and sentiment analysis techniques," *International Journal of Soft Computing (IJSC)*, vol. 5, pp. 11–21, 02 2014.

[17] A. Kumar and A. Jaiswal, "Empirical study of twitter and tumblr for sentiment analysis using soft computing techniques."

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.