An implementation of support vector machine on sentiment classification of movie reviews

I M Yulietha, S A Faraby, Adiwijaya, and W C Widyantybas
School of Computing Telkom University

irenemys@student.telkomuniversity.ac.id, saidalfarby@telkomuniversity.ac.id, adiwijaya@telkomuniversity.ac.id, windawinda@student.telkomuniversity.ac.id

Abstract. With technological advances, all information about movie is available on the internet. If the information is processed properly, it will get the quality of the information. This research proposes to classify sentiments on movie review documents. This research uses Support Vector Machine (SVM) method because it can classify high dimensional data in accordance with the data used in this research in the form of text. Support Vector Machine is a popular machine learning technique for text classification because it can classify by learning from a collection of documents that have been classified previously and can provide good result. Based on number of datasets, the 90-10 composition has the best result that is 85.6%. Based on SVM kernel, kernel linear with constant 1 has the best result that is 84.9%

1. Introduction

Information needs are driven by the abundance of technology and research. There is so much information, so it necessary to present information but not lose the value of the information [1]. With the development of web, more people that can write their opinion about a product or service. Some of great website allows the user to contribute, modify, and increase the content [2]. Every internet user can express their opinion about something freely. One of them is movie review.

Nowadays everyone can give their opinion in social network [3] [4]. As well as movie review [4]. By the technology, information about movie review are available on internet. It hard to find the correct information that user wants. If the information is processed we can get the value and the quality of the information. It will allow users to get movie quality easily by movie review classification.

From this movie review, there is a challenge to handle negation word in review. Negation word is statement that has denial or reverses the value. Usually negation word uses ‘not’. Negation word may affect movie review because it can change the value of sentiment. Positive words can turn negative by negation word.

Sentiment analysis used to see the opinion or tendency of opinion about a movie is positive or negative. This sentiment analysis research was built using a machine learning approach known as Support Vector Machine and devoted to movie review in English language. Machine learning is used because it can be used in every area of computerization such as text analyze, image processing, finance, search and recommendation engine, speech understanding, etc. Machine learning in this case is used to classify texts based on positive or negative user opinion.
2. Literature Review

There are several research that have been done in sentiment analysis, some of them are Hidayatullah and Azhari in 2014 by using Naïve Bayes Classifier and Support Vector Machine to analyze data tweet of presidential vote in 2014. Support Vector Machine have better result than Naïve Bayes. Songbo Tan et al (2008) compared Naïve Bayes, centroid classifier, K-Nearest Neighbor (KNN), winnow classifier and Support Vector Machine on dataset in China language. The result is Support Vector Machine have the best result. By those research Support Vector Machine have better accuracy, so the author uses Support Vector Machine to classify movie reviews.

Another research that have been conducted in sentiment analysis classification is Maraes et al that has analyzed about movie review and product from Amazon used Support Vector Machine and artificial Neural Network. Fatimah Wulandini and Anto Satriyo Nugroho have analyzed Text Classification using Support Vector Machine for Web Mining Based Spation Temporal Analysis of the Spread of Tropical Disease by comparing four classify methods. Zhang et al in 2011 has analyzed sentiment classification in restaurant reviews on internet in Canton language using Naïve Bayes and Support Vector Machine. Setyo Budi in 2017 has analyzed about movie review from Cornell Computer Science as dataset using K-Means method. There are several classification method commonly used for text classification or speech classification, that is Multinomial Naïve Bayes [5], Bayesian Network [6], and Hidden Markov Model [7] [8].

Sentiment analysis is one of computer science to identify emotion, assessment, attitude, opinion, evaluation of a product, services, organization, individual, public figure, topic, event, or other activity. The goal is to categorize someone opinion as a sentiment positive or sentiment negative [12]. In this paper, we propose a method for classifying sentiments on the document movie reviews. The dataset source is https://www.cs.cornell.edu/people/pablo/movie-review-data/. From that source, there are 1000 reviews with positive sentiment and 1000 reviews with negative sentiment. Data has been labeled positive and negative [9].

Support Vector Machine (SVM) that developed by Boser and Guyon, Vapnik in 1992 is a classification technique that aims to find the hyperplane with the largest margin [20]. The good hyperplane is obtained with the maximal of margin value. Margin is a distance between hyperplane with support vector (closest point with hyperplane).

3. System Design

In this research, we will build system that can classify sentiment on movie review. The system described in figure 1.
The purpose of this research is to analyze:

a. Comparison of the amount of training data and testing data to classification model

b. Influence of pre-processing and Negation Handling to Support Vector Machine method in movie review

In validation stage is used k-fold cross validation method. In k-fold cross validation data divided into k parts, that is D₁, D₂, …Dₖ and each D has same quantity of data. It will be iterated k times, which in each i iteration will be specified as testing data. Another data will be used as training data to get the classification model. Evaluation is to measure the performance of the system. The result will be evaluated by predictive classification table, the Confusion Matrix. Confusion Matrix [22] is table that can be used to determine performance of classification model [23]. The example of confusion matrix can be seen in Table 1.

Table 1 Predictive Classification.

| Predicted Class | Actual Class |
|-----------------|--------------|
|                 | +            | -            |
| +               | True Positive| False Positive|
| -               | Negative True| False Negative|

From table 1, there is document categorize in a searching process [24]:
1. True Positive (TP) is system prediction result is positive and the target result is positive.
2. True Negative (TN) is system prediction result is negative and the target result is negative.
3. False Positive (FP) is system prediction result is positive but the target result is negative.
4. False Negative (FN) is system prediction result is negative but the target result is positive.

4. Data and Analysis

a. Influence of Number of Dataset
Based on the result, more number of data given to the training data, the accuracy higher. That is because the classification model can handle more diversity of testing data so the model can classify better. Testing result with composition data between training and testing 90:10 have higher accuracy, ie 85.6% with 1800 training data and 200 testing data. In composition 50% training data and 50% testing data has smallest value of F1-Score that is 82.7%, the number of training data is less than before so the classification model can’t handle more diversity on training data but because of good training data characteristic (data have difference to classified) so the model that have been built is good.

b. Influence of SVM Kernel Function
Comparison of SVM kernel uses three different kernels. That is linear, Radial Basic Function (RBF) and polynomial. The results are given in the Table 2.

| No. | C Value | F1-Score Average (%) |
|-----|---------|----------------------|
| 1.  | 0.01    | 32.7%                |
| 2.  | 0.1     | 38.3%                |
| 3.  | 1       | 84.9%                |
| 4.  | 10      | 83.6%                |
| 5.  | 100     | 83.3%                |
| 6.  | 1000    | 83.7%                |

Based on the experiment. Table 2 shows that the comparison of accuracy values on SVM Linear with C as constant, F1-Score 84.9% with C=1.0 is the best result. In this scene, C affects the accuracy of testing data classification. Greater the value of C then accuracy result is smaller. This is because trade-off between margin and error is bigger. The value of results is obtained from average of all fold.

Table 3. RBF Kernel Result.

| C     | 0.01 | 0.1 | 1  | 10 | 100 | 1000 |
|-------|------|-----|----|----|-----|------|
| 0.01  | 32.4%| 32.2%| 32.1%| 32.8%| 32.4%| 32.6%|
| 0.1   | 32.6%| 32.5%| 32.4%| 32.6%| 32.8%| 32.2%|
| 1     | 32.5%| 79.7%| 83.6%| 33.5%| 33.4%| 32.6%|
| 10    | 79.3%| 83.8%| 83.9%| 32.7%| 32.8%| 32.7%|
| 100   | 83.3%| 83.8%| 83.8%| 32.7%| 32.7%| 32.7%|
| 1000  | 83.6%| 83.5%| 84.3%| 33.4%| 32.9%| 32.2%|

Based on the experiment. Table 3 shows that F1-Score from RBF Kernel with C and γ, the best result of kernel RBF classification is 84.3% with C = 1000 and γ = 1.0. The value of C and γ can affect accuracy of testing data classification. The value of C is the most commonly used in all SVM kernels. The function of γ is to determine proximity level between two points so it easier to find the consistent hyperplane. Based on the result, the smaller value of gamma then the F1-Score value is smaller so it needs larger C parameter.
Table 4. Polynomial Kernel Result.

| d | C   |   |   |   |   |
|---|-----|---|---|---|---|
|   | 0.01| 0.1| 1 | 10 | 100 | 1000 |
| 1 | 32.5% | 32.8% | 32.7% | 32.7% | 32.1% | 31.8% |
| 2 | 32.2% | 32.3% | 32.4% | 32.7% | 32.7% | 32.6% |
| 3 | 32.2% | 32.2% | 32.7% | 32.1% | 32% | 31.4% |

Based on the experiment. Table 4 shows that the F1-Score value of polynomial kernel classification with C and d (degree). The best result of polynomial kernel classification was 32.8% with d = 1 and C = 0.1. For the optimal result d may change. Based on the character of the data, in this testing data d=1 optimal result. The function of d is to help mapping data from input space to higher dimension in feature space, so in the new dimension can be found a consistent hyperplane.

c. Influence of Pre-processing and Negation Handling

In this research, pre-processing step is to compare between stemming and lemmatization. Stemming and lemmatization have different word-cutting processes. Stemming works by eliminate any prefixes, inserts, suffixes, or combination of prefix and suffix. But lemmatization works by change each word to the base of word.

Table 5 Lemmatization and Stemming Result.

| Fold | F1-Score |     |     |
|------|----------|-----|-----|
|      | Lemmatization | Stemming |
| 1    | 85.8%    | 84.1% |
| 2    | 84.8%    | 83.9% |
| 3    | 84.1%    | 84.8% |
| 4    | 83.6%    | 83.8% |
| 5    | 86.2%    | 83.1% |
| Average | 84.9%    | 83.9% |

Based on the experiment. Table 5 shows that lemmatization value is higher than stemming. Lemmatization has better value because each word is converted into base word and make the true feature in each document increase, that cause the extracted feature increase too. This research suggests that lemmatization tends to produce better classification performance. Another reason that causes stemming process has less feature than lemmatization because there are difference of word cutting process. Evaluation result of stemming is reduced because there is missing information from the word that has been stemmed.

In this research is comparing the influence of using negation handling and without negation handling. This process finds negation word, the example are no, not, etc. If in a sentence contain negation word (no or not), then words after negation word are marked as negation word.

Based on the experiment. Table 6 shows that the best value of F1-Score is by using negation handling, that is 84.4%. Without using negation handling the F1-Score is 82.4%. The difference between negation handling and without negation handling is 0.024. Negation handling can increase performance of sentiment analysis in sentences. The performance increase because of a word in different sentences context can detect negation synthetic that may affect the class prediction result on movie review that used.
Table 6. Influence of Negation Handling

| Fold | Without Negation Handling | Using Negation Handling |
|------|---------------------------|-------------------------|
|      | Precision | Recall | F-1 Score | Precision | Recall | F-1 Score |
| 1.   | 79.4%     | 81.1%  | 80.3%     | 86.5%     | 78.1%  | 82.1%     |
| 2.   | 85.2%     | 79%    | 82.4%     | 84.8%     | 89.6%  | 87.2%     |
| 3.   | 85.4%     | 83.2%  | 84.3%     | 87.5%     | 85.5%  | 86.5%     |
| 4.   | 85.1%     | 87.3%  | 83.4%     | 85.4%     | 86.6%  | 86%       |
| 5.   | 83.6%     | 79.8%  | 81.7%     | 80.6%     | 84%    | 82.2%     |
|      | **Average** | **82.4%** | **84.8%** | **82.4%** | **84.8%** | **84.8%** |

5. Conclusion
Base on the research, the conclusion are more data that used as training data, the F1-Score result is better for classify. In this case the best result is 85.6% with 90% data as training data and 10% data as testing data. In linear separable and non-linear separable testing, the better F1-Score is 84.9% using linear separable. In this case used linear kernel, RBF kernel, and polynomial kernel in SVM. The result is linear kernel has the best result of F1-Score is 84.9%. Using negation handling give effect in classification. Classification with negation handling has better result. In stemming and lemmatization testing, lemmatization F1-Score is better than stemming because the difference in process cut-off word. For the next research, the different selection feature is expected to improve the accuracy result for the example on polynomial accuracy result.

References
[1] B. Liu, "Sentiment Analysis: A Multi-Faceted Problem," IEEE Intelligent Systems, 2010.
[2] K. Yessenov and S. Misailovic, "Sentiment Analysis of Movie Review Comments," 2009.
[3] C. L. Liu, W. Hsaio, C. H. Lee and E. Jou, "Movie Rating and Review Summarization in Mobile Environment," IEEE Transactions on Systems Man, and Cybernetics, Part C (Applications and Reviews), 2012.
[4] V. Chandani, R. S. Wahono and Purwanto, "Komparasi Algoritma Klasifikasi Machine Learning Dan Feature Selection pada Analisis Sentimen Review Film," Journal of Intelligent Systems, vol. 1, 2015.
[5] R. A. Aziz, M. S. Mubarok and Adiwijaya, "Klasifikasi Topik Pada Lirik Lagu Dengan Metode Multinomial Naive Bayes," Indonesia Symposium on Computing (IndoSC), 2016.
[6] A. H. R. Z.A, M. S. Mubarok and Adiwijaya, "Learning Struktur Bayesian Networks Menggunakan Novel Modified Binary Differential Evolution Pada Klasifikasi Data," Indonesia Symposium on Computing (IndoSC), 2016.
[7] I. N. Yulita, H. L. The and Adiwijaya, "Fuzzy Hidden Markov Models For Indonesian Speech Classification," JACIII, vol. 16(3), pp. 381-387, 2012.
[8] U. N. Wisesty, M. Mubarok and Adiwijaya, "A Classification Of Marked Hijaiyah Letters' Pronunciation Using Hidden Markov Model," AIP Conference Proceedings 1867, vol. 020036, 2017.
[9] E. Simoudis, "Reality Check for Data Mining," IEEE Expert: Intelligent Systems and Their Application, vol. 11, 1996.
[10] M. Hu and B. Liu, "Mining and Summarizing Customer Review," 2004.
[11] Irawati, "Optimisasi Parameter Support Vector Machine (SVM) Menggunakan Algoritme Genetika," 2010.
[12] B. Liu, "Sentiment Analysis And Opinion Mining," 2012.
[13] R. Feldman and J. Sanger, "The Text Mining HandBook: Advanced Approaches in Analyzing Unstructured Data," 2007.
[14] M. S. Mubarok, Adiwijaya and M. D. Aldhi, "Aspect-based Sentiment Analysis to Review Products Using Naive Bayes," AIP Conference Proceedings 1867, vol. 020060, 2017.
[15] C. C. Aggarwal and C. Zhai, "A Survey of Text Clustering," IBM T.J. Watson Research Center, pp. 78-128, 2012.
[16] Adiwijaya, Aplikasi Matriks dan Ruang Vektor, Yogyakarta: Graha Ilmu, 2014.
[17] Adiwijaya, Matematika Diskrit dan Aplikasinya, Bandung: Alfabeta, 2016.
[18] J. Han and M. Kamber, "Data Mining: Concepts and Techniques," 2006.
[19] O. Maimon and L. Rokach, "Data Mining and Knowledge Discovery Handbook," 2010.
[20] P.-N. Tan, M. Steinbach and V. Kumar, Introduction to Data Mining, Boston: Pearson Addison Wesley, 2006.
[21] A. S. Nugroho, A. B. Witarto and D. Handoko, Support Vector Machine - Teori dan Aplikasinya dalam Bioinformatika, Penerbit IlmuKomputer.Com, 2003.
[22] D. Xhemali, C. J. Hinde and R. G. Stone, "Naive Bayes vs. Decision Trees vs. Neural Networks in the Classification of Training Web Pages," International Journal of Computer Science, no. 4(1), pp. 16-23, 2009.
[23] L. Hamel, The Encyclopedia of Data Warehousing and Mining, 2nd Edition, Idea Group Publishers, 2008.
[24] S. Bird, E. Klein and E. Loper, Natural Language Processing with Python, 2009.
[25] N. M. S. Hadna, P. I. Santosa and W. W. Winarno, "Studi Literatur Tentang Perbandingan Metode Untuk Proses Analisis Sentimen Di Twitter," Seminar Nasional Teknologi Informasi dan Komunikasi 2016 (SENTIKA 2016), 2016.
[26] P. Chaovalit and L. Zhou, "Movie Review Mining: a Comparison between Supervised and Unsupervised Classification Approaches," Proceedings of the 38th Hawaii International Conference on System Sciences, 2005.