Prediction of Movie Sentiment based on Reviews and Score on Rotten Tomatoes using SentiWordnet

Suhariyanto
Faculty of computer science
Dian Nuswantoro University
Semarang, Indonesia
suhariyanto@dsn.dinus.ac.id

Ari Firmanto
Department of Informatics
Institute Technology Sepuluh
Nopember Surabaya, Indonesia
arifirmanto.17051@mhs.its.ac.id

Riyanarto Sarno
Department of Informatics
Institute Technology Sepuluh
Nopember Surabaya, Indonesia
riyanarto@if.its.ac.id

***Abstract*— With the number of films released each year, the movie review website is becoming more popular. One of the most referenced movie review websites is Rotten Tomatoes. Rotten Tomatoes recommend films based on their Tomatometer. Tomatometer represents the percentage of professional critic reviews that are positive or negative for a given film or television show. While fresh reviews represent positive sentiment, rotten reviews mean that the movie critics give the movie negative sentiments. Unfortunately, the method to determine the given score is not available to the public. Thus, the public does not know which parameter affect the prediction of the sentiment. This paper proposes a new method to predict the sentiment of the movie on the rotten tomatoes by combining the sentiment score from SentiWordnet and expert original score. the result of the experiment shows that the proposed method gives better F measure compared to those of the other methods with the value of 0.97.**

keywords—rotten tomatoes, sentiment analysis, sentiwordnet

I. INTRODUCTION

In the era of information technology, any information can be gained quickly such as movie information. Every year a new movie is released, it will be better if there is a recommendation to filter movie that people wants to watch. Nowadays, there are a lot of movie recommendations, one of them is rotten tomatoes. It gives the recommendation based on the score that calculates by the system using expert critic review. However, the process behind it is not publicly available.

Sentiment analysis is one of the methods to analyses user opinion from the review. From the previous research, many researchers using two approaches in sentiment analysis: lexical based approach and supervised machine learning approach. The Lexical approach using the dictionary that has sentiment polarity on them, such as SentiWordnet. Supervised machine learning approach using statistical analysis to predict sentiment polarity using the defined label as a target. The problem of supervised machine learning is dependent on the domain. If data set in the different domain then, machine learning need to be retrained to get high accuracy [1].

SentiWordnet [2] is a dictionary that has sentiment polarity based on Wordnet. Wordnet structure is a tree-like, that start with root word as hypernym and have branched from the root that called hyponym. It uses a random walk algorithm to give a score from Wordnet. To choose correct sense form synset (synonym set), word sense disambiguation algorithm is needed, because wordnet structure has the different meaning for the same word.

In RottenTomatoes, sentiment polarity is classified into two classes, rotten and fresh. Rotten is a negative sentiment and fresh is a positive sentiment. An expert does not only give their review, but they also give their rating called original score. The proposed method of this paper is to combined both expert review and expert original score. SentiWordnet used to extract sentiment score from review. To combined with the original score, normalization is needed for both values. The result of this is a feature that will be proceeded by logistic regression as a classification algorithm to predict movie sentiment.

II. RELATED WORK

Pang and Lee [3] used Naive Bayes, Maximum Entropy (ME), and SVM to classify binary sentiment on movie reviews. They collected the dataset from IMDB and used various features which were highest results come from SVM with unigram feature and got accuracy about 82.9%.

Dang et al [4] classified sentiment using SVM for a multi-domain dataset. They collected 305 positive reviews and 307 negative reviews from the digital camera. Information gain was used as a feature selection and got accuracy for 84.15%.

Agarwal et al. [5] classified sentiment from tweets using SVM (Support Vector Machine). They experimented with 11875 manually labeled tweets annotation. The authors used the combination of features which were unigram, sentiment score, and dependency tree. The highest accuracy of the model is 75.39%.

Xia et al. [6] used ensemble machine learning model to classified sentiment. The classifiers were Naive Bayes, Maximum Entropy, and SVM. The features that used was POS Tagging and word relation. Ensemble classifier gave the highest accuracy about 87.7%.

Pamungkas and Putri [7] used SentiWordnet to classified play store review dataset. They used first sense to choose synset from wordnet. The result of their experiment is about of 67.93%. Fika et al [8] used corpus-based emotion (CBE) to predict music emotion based on lyrics. CBE is the merging of two computational model which are WNA (Wordnet Affect Emotion) and ANEW (Affective Norms for English Words) using terms similarity, distancing of the node, and topic modeling [9].

Bagus et al [10] extracted aspect from product review and determined sentiment of each aspect. The Dataset used was product review from Amazon. The authors handled both local and global context from the review to improve sentiment analysis. In the field of business model, Endang et

al [11] built a lexical database using the same method with BabelNet for business process, it called B-BabelNet. BabelNet [12] is the lexical database of semantic network based on Wikipedia and Wordnet. B-BabelNet calculated semantic similarity in the domain of business process and it r caould be used to improve sentiment analysis in the business process.

III. RESEARCH METHOD

In this paper, the method is divided into crawling the data from Rotten Tomatoes, text preprocessing, feature extraction and sentiment classification. Detail processes can be seen in Figure 1, which is described as follows.

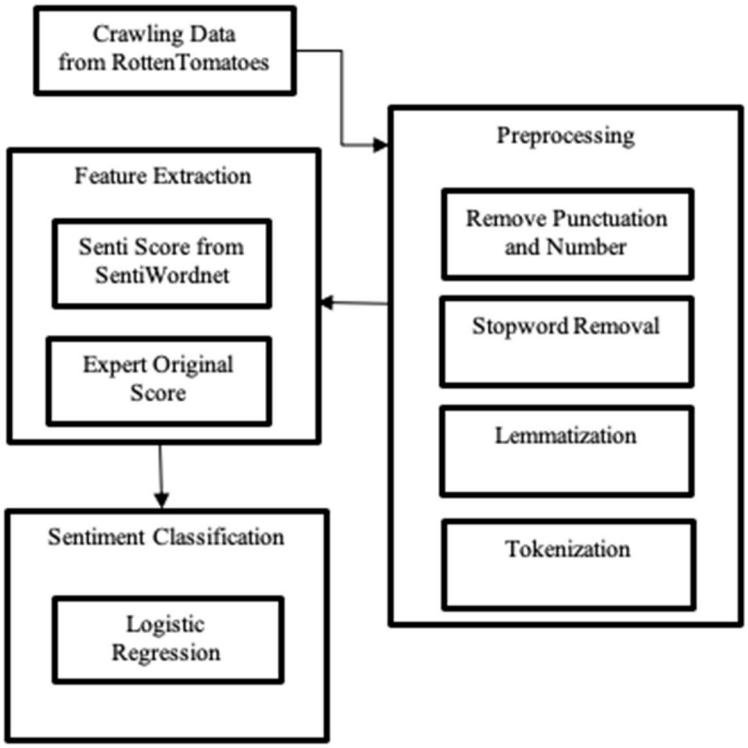


Figure 1. Propose Framework

A. Dataset Collection

Data is collected by crawling on rotten tomatoes. It should have an expert original score if it does not have then data is skipped. Then data is saved into CSV file. The total amount of data is 300, which were evenly selected with 150 data for rotten class and 150 data for fresh class. This balanced selection was carried out on purpose to get better result.

The purpose of the preprocessing is to prepare data and to remove noise from the data in order to increase of the accuracy for machine learning to learn pattern from the data. The steps are:

- **Remove Punctuation and Number** is the step to remove punctuation and number from the movie reviews.
- **Normalize into Lower Case** is the step to change all word into lower case.
- **Stop Word Removal** is the step to remove English stop word from text review. NLTK library is used for stop word removal.
- **Lemmatization** is the step to change a word into a lemma using the dictionary. In this paper, NLTK library used to change a word into a lemma.
- **Tokenization** is the step to change text review into the token of words.

C. Feature Extraction

The Feature that used for feature extraction is a sentiment score from SentiWordnet and expert original score. The detail process for each feature is:

- SentiWordnet

Score calculated using SentiWordnet. In this paper, only an adjective score and verb score used. SentiWordnet modeled part of speech (POS) from the treebank parser. Thus, score collected only if that word has POS tagging JJ, JJR, JJS, VB, VBD, VBG, VBN, VBP, and VBZ. The detail treebank parsers can be seen in Table 1.

TABLE I. TREE BANK PARSER

Tree Bank Parser	Definition
JJ	adjective
JJR	adjective, comparative
JJS	adjective, superlative
VB	Verb, base form
VBD	Verb, past tense
VBG	Verb, gerund or present participle
VBN	Verb, past participle
VBP	Verb, non 3rd person singular present
VBZ	Verb, 3rd person singular present

To find the correct synset from SentiWordnet, adapted weighted graph [13] and lesk algorithm [14] are used. Sentiment score is calculated by the sum of the positive or negative score divided by the total of the positive and negative score. The equation can be seen in Equation 1, 2, 3, and 4.

$$pos\_score = \sum_{i=i}^{n} pos\_score\_senti(i) \quad (1)$$

$$neg\_score = \sum_{i=i}^{n} neg\_score\_senti(i) \quad (2)$$

$$total\_score = pos\_score + neg\_score \quad (3)$$

$$senti\_score = \begin{cases} \frac{pos\_score}{tota\_score} if\ pos > neg \\ \frac{neg\_score}{total\_score}\ if\ neg > pos \end{cases} \quad (4)$$

In the equation above, n is the total word from text review, pos_score_senti is individual word score from SentiWordnet, pos_score is the total positive score, neg_score_senti is individual word score from SentiWordnet, neg score is the total negative score, and senti_score is the final score used as a feature.

- Expert Original Score

For the expert original score, the value divided into categorical and numeric value because some expert use alphabet to give a score in their review. For categorical, the self-define rule used for mapping categorical into numeric value. The rule can be seen in Table 2

TABLE II. NORMALIZED CATEGORICAL SCORE

Expert Score	Transform Score
A	5
B	4
C	3
D	2
E	1
+	added by 0.25
-	subtract by 0.25

For the numerical score, must be normalized first because it contained difference range of value. The equation can be seen in Equation 5:

$$rating = 5 / max\_org\_rating * org\_rating \quad (5)$$

In the Equation 5, max_org_rating is the maximum range rating from the expert original score and org_rating is the original score from the expert. After rating has obtained, then it will be divided by 5 which are the maximum rating score. This equation can be seen in Equation 6.

$$exp\_final\_rating = rating / 5 \quad (6)$$

D. Sentiment Classification

For sentiment classification, data will be modelled into the regression, where the goal is to predict the weight between sentiment score and expert original score. The weight implies the important features between them. Logistic regression is used in this paper, because the target is binary classification. The equation for logistic regression can be seen in Equation 7:

$$log\_y = \frac{exp(\alpha * X1 + \beta * X2 + \gamma)}{1 + exp(\alpha * X1 + \beta * X2 + \gamma)} \quad (7)$$

Where γ is the bias, α and β is the weight, and X1, X2 is the features. To find the α and β that satisfy Equation 7 that minimizes the error, stochastic gradient descent used. The stochastic gradient descent is the variant of gradient descent that used only randomly one piece of the data from dataset in the calculation. The equation for stochastic gradient descent can be seen in Equation 8 and 9.

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^i log\_y + (1 - y^i) log(1 - log\_y)] \quad (8)$$

$$\theta_j = \theta_j - \alpha J(\theta) \quad (9)$$

m is the total sample size, y is the ground truth of the rating data, and α is the learning rate.

IV. RESULT

To evaluate the results, the proposed method was compared with the baseline method. The baseline method that used in this paper is SentiWordnet, expert original score, and SVM (Support Vector Machine). SentiWordnet sentiment orientation calculated using Equation 1 and 2. If the positive score greater than the negative score than it classified as fresh otherwise is rotten. For the expert original score, to determined sentiment orientation threshold is used. The threshold value is 3 if the original score greater than or equal 3 then classified as fresh otherwise is rotten. And SVM using only Term Frequency-Inverse Document Frequency (TF-IDF) features with linear kernel. For evaluation, the confusion matrix will be used. The equation can be seen in Equation 10, 11 and 12:

$$precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (10)$$

$$recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (11)$$

$$F - measure = 2 * \frac{precision * recall}{precision + recall} \quad (12)$$

For SentiWordnet and expert original score, evaluation matrix will use all data because it does not have a model that needs to be trained. Otherwise, for the proposed method and SVM data will be split into training and testing to overcome overfitting on the data. Distribution of the data can be seen in Figure 2. From Figure 2, the total of rotten and fresh from SentiWordnet and Expert Original Score is 150. For SVM and the proposed method, the total of data is 29 for rotten and 31 for fresh. This is because it is using only testing data for the evaluation matrix. Confusion matrix comparation can be seen in Figure 3 and Table 3.

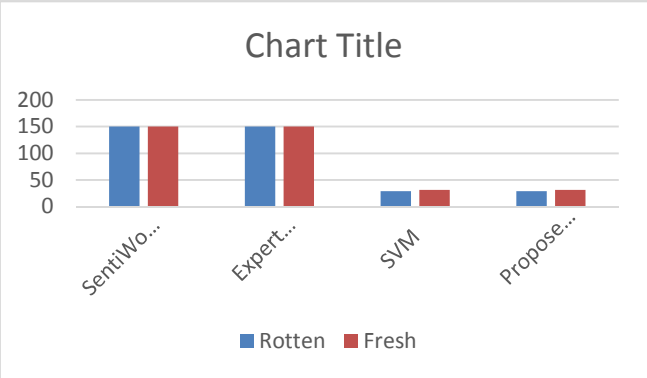


Figure 2. Data Distribution of Each Method

TABLE III. AVERAGE PRECISION, RECALL, AND F-MEASURE COMPARATION

Method	Precision	Recall	F Measure
SentiWordnet	0.51	0.51	0.51
Expert Original Score	0.92	0.91	0.91
SVM - TFIDF	0.77	0.77	0.77
Proposed Method	0.97	0.97	0.97

TABLE IV. PRECISION, RECALL, AND F-MEASURE COMPARATION OF FRESH SENTIMENT

Method	Precision	Recall	F Measure
SentiWordnet	0.51	0.52	0.51
Expert Original Score	0.86	0.98	0.92
SVM - TFIDF	0.74	0.79	0.77
Proposed Method	1.0	0.93	0.96

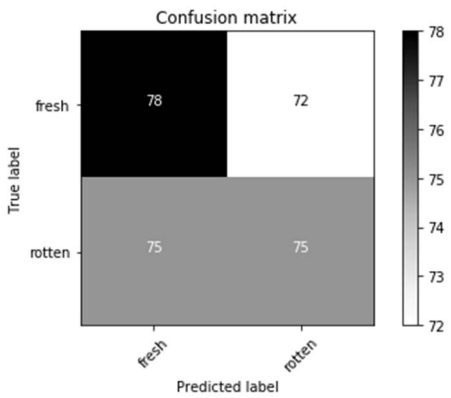
TABLE V. PRECISION, RECALL, AND F-MEASURE COMPARATION OF ROTTEN SENTIMENT

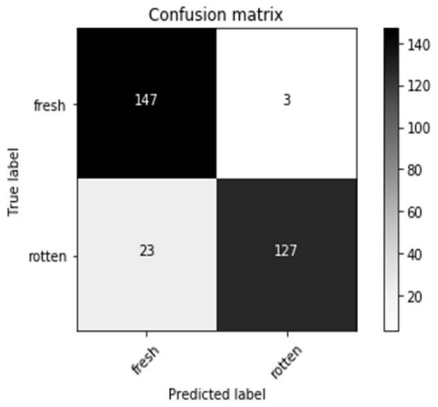
Method	Precision	Recall	F Measure
SentiWordnet	0.51	0.50	0.51
Expert Original Score	0.98	0.85	0.91
SVM - TFIDF	0.79	0.74	0.77
Proposed Method	0.94	1.0	0.97

From Table 3, our proposed method outperformed other methods with F1 measure 0.97. For SVM, using TF-IDF features depends on the word distribution. If the expert using various of words to describe their review, then data matrix will be sparse and good accuracy will be difficult to achieve. For the proposed method, using the combination of SentiWordnet and expert original score show good result. SentiWordnet cannot handle the review that has implicit sentiment, for instance in review: "*Maybe the quips and the punch-ups are there because they have to be, what with the film's steady parade of failure and even death. Plans Fail. Character fails. Even sacrifices fail. It's not exactly refreshing, but it is bracing, and even gratifying*". In that review, there are a lot of words that fail indicated to negative sentiment, for example is *failure*, *fail*, and *death*. Thus, using SentiWordnet will be classified as rotten but the ground truth is fresh. The expert original score is the second highest result from the baseline, this is because original score is the feature importance. Table 6 shows the importance of features from the trained model, and the negative value means that the corresponding feature pushes the classification more towards the negative class in this case, is rotten. Thus, the recall of the rotten class is higher than fresh class, this can be seen in Table 4 and Table 5. Thus, using only expert original score baseline good result can be achieved. But, the expert original score has lower precision than the proposed method. This can be seen from review "*Ultimately, it is more interesting to think about than it is to watch*" the expert gives rating 2 from 5, then it will be classified into rotten, but the ground truth is fresh. Our proposed method fixes the problem above and gets the closest result with rotten tomatoes ground truth.

TABLE VI. FEATURE IMPORTANCES OF LOGISTIC REGRESSION MODEL

	Sentiment Score	Expert Original Score
Coefficient	0.289	-2.313





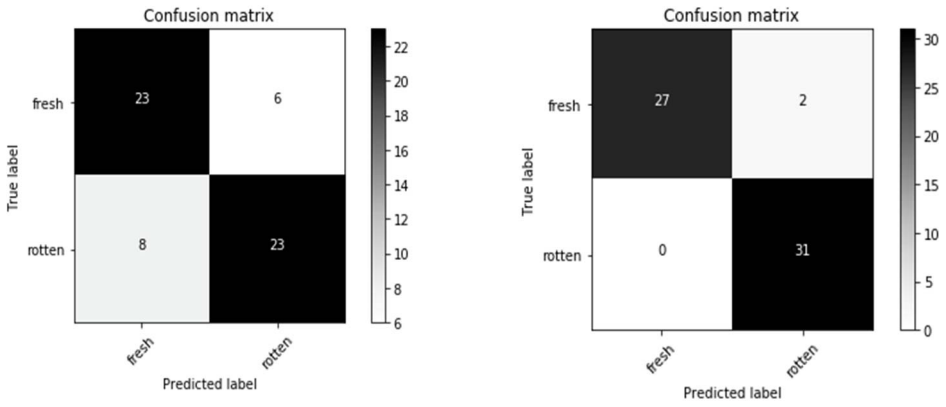


Figure 3. Top left is the Confusion Metrix of SentiWordnet, top right is the Confusion Metrix of Expert Original Score, bottom left is the Confusion Metrix of SVM using TF-IDF feature, and bottom right is the Confusion Matrix of the proposed method.

V. Conclusion and Future Work

The experiment showed that there were many implicit reviews from the expert that SentiWordnet had failed to recognize it, whereas the proposed method could recognize implicit reviews because of the additional features; i.e., sentiment score and expert original score.

For future work, we will implement imbalance algorithm to handle imbalance class. Also, the effect of incomplete data of expert original score will be analyzed.

References

[1] B. Liu, "Sentiment Analysis and Opinion Mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[2] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0 : An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining SentiWordNet," *Analysis*, vol. 10, pp. 1–12, 2010.

[3] B. Pang, L. Lee, and S. Vaithyanathan, "{T}humbs up? {S}entiment classification using machine learning techniques," in *Proc. of the EMNLP'02*, 2002.

[4] Y. Dang, Y. Zhang, and H. Chen, "A lexicon-enhanced method for sentiment classification: An experiment on online product reviews," in *IEEE Intelligent Systems*, 2010, vol. 25, no. 4, pp. 46–53.

[5] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of Twitter data," *Association for Computational Linguistics*, pp. 30–38, 2011.

[6] R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," *Information Sciences*, vol. 181, no. 6, pp. 1138–1152, 2011.

[7] E. W. Pamungkas and D. G. P. Putri, "An experimental study of lexicon-based sentiment analysis on Bahasa Indonesia," in *Proceedings - 2016 6th International Annual Engineering Seminar, InAES 2016*, 2017, pp. 28–31.

[8] F. Hastarita Rachman, R. Sarno, and C. Fatichah, "Music Emotion Classification based on Lyrics-Audio using Corpus based Emotion," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 3, pp. 1720–1730, 2018.

[9] F. H. Rachman, R. Sarno, and C. Fatichah, "CBE: Corpus-based of emotion for emotion detection in text document," in *Proceedings - 2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering, ICITACEE 2016*, 2017, pp. 331–335.

[10] B. S. Rintyarna, R. Sarno, and C. Fatichah, "Enhancing the performance of sentiment analysis task on product reviews by handling both local and global context," *International Journal of Information and Decision Science*, vol. 11, 2018.

[11] E. W. Pamungkas, R. Sarno, and A. Munif, "B-BabelNet: Business-Specific Lexical Database for Improving Semantic Analysis of Business Process Models," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 15, no. 1, p. 407, 2017.

[12] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," *Artificial Intelligence*, vol. 193, pp. 217–250, 2012.

[13] B. S. Rintyarna and R. Sarno, "Adapted weighted graph for Word Sense Disambiguation," in *2016 4th International Conference on Information and Communication Technology, ICoICT 2016*, 2016.

[14] S. Banerjee and T. Pedersen, "An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet," *International Conference on Intelligent Text Processing and Computational Linguistics*, vol. 2276, pp. 136–145, 2002.