Sentiment Classification of Film Reviews Using IB1

Oswin Rahadiyan H
Informatics Engineering
DWCU
Yogyakarta, Indonesia
oswin.rh@ti.ukdw.ac.id

Gloria Virginia
Informatics Engineering
DWCU
Yogyakarta, Indonesia
virginia@staff.ukdw.ac.id

Antonius Rachmat C
Informatics Engineering
DWCU
Yogyakarta, Indonesia
anton@ti.ukdw.ac.id

***Abstract* - Review of an object or product is important to public judgment of the product. Review can be used in film industry to consider a movie is worth to watch or not. Sentiment classification is used to detect the class of a commentary or review. The purpose of this research is to classify film reviews from Rotten Tomatoes using text mining methods. Classification methods are various like Naive Bayes, Instance Based Learning, Decision Tree, SVM. And IB1 (Instance Based Learning 1) is used on this research because of its simplicity and accuracy. WordNet component also used to expand the similar words on database. Performance of the algorithm is measured by evaluation methods such as Accuracy, Precision, Recall and F-measure.**

Keywords - classification; film review; IB1; WordNet

I. INTRODUCTION

In text classification, there are various algorithms to classify text data, e.g. Decision Trees, Naive Bayes, Support Vector Machine, Instance-Based Learning and many more [3]. This study uses Vector Space model to implement Instance-Based Learning: IB1 which is using TF-IDF weighting scheme and Cosine similarity function. The objects on this study are movie reviews because a review can influence people's assumption [7]. Instance Based Learning is a simple yet effective classification method [1]. This research aims to implement IB1 algorithm effectively on a sentiment classification system that will categorize the review in two classes, namely positive review or negative review. And see the results of the classification and evaluation based on the use of WordNet. Data or text corpus used in this study comes from Rotten Tomatoes website.

II. LITERATURE STUDY

IB1 classification method was developed in 1991 as the first of IBL (Instance Based Learning) family and development of Nearest Neighbor algorithm [1].

The comparative study of classification accuracy performance of various data sets with different classification methods such as Naive Bayes, Bayesian Networks, LBR and IBL showed that the best classification result was the one with Naïve Bayes or IBL classifier [2]. The result showed that IBL has the fastest time to build the classification model compared with the other method.

Another support study by Vijayarani in 2013, a research of text classification using Bayesian (BayesNet, Naïve Bayes) and Lazy (k-NN, K *, IBL) algorithms. The highest classification result can be achieved by using the Lazy methods (k-NN and IBL) [9].

Previous studies of sentiment classification have implemented lexical resource for opinion mining called SentiWordNet [6][8]. On this research, WordNet is used as lexical component with detection of negative token as modified and specific preprocessing rule.

IB1 is used to classify text sentiment because it shows great performances from the research done before. And this research focuses on how to classify documents according to the sentiment class or overall opinion towards an object (positive or negative) with the implementation of WordNet database. WordNet's implementation is observed and evaluated to aim better classification result. This research also implements certain preprocessing rule, i.e. detection of negative token. The purpose of using the negative token detection is to merge words into one entity e.g. don't care, doesn't matter, never again, not good, isn't boring.

III. BACKGROUND

A. Preprocessing

The objectives of data preprocessing is to normalize the document's format into structural format [4]. Preprocessing stages of this research are lowercase the document, tokenization, stopword removal, TF-IDF weighting and WordNet implementation.

2166-0670/16 $31.00 © 2016 IEEE
DOI 10.1109/ISMS.2016.38

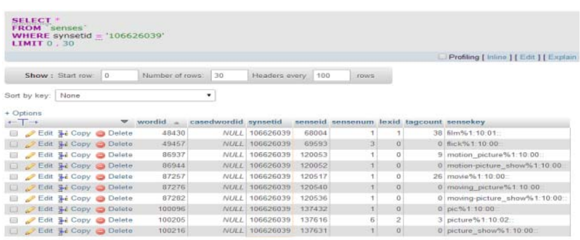
B. Detection of Negative Token

Relevant tokens or words are the key of good classification. If tokens are noisy, they will affect the accuracy of classification. In this case, not all of the stop words are noises. Some words are needed to improve the classification performance. This preprocessing rule is issued to determine some negative words as one entity. Negative tokens refer to these examples, e.g. don't, doesn't, never, not, less, no. If system found negative token then token will be merged with the next token e.g. don't know, doesn't matter, never again, not good which is supposed to be one entity.

C. WordNet

WordNet is a free lexical database in the field of computational linguistics developed by Princeton University[1]. In this study, WordNet database is used as a support component for synonymous tokens. Words like movie and film have a same definition. Synonymous token will be expanded into enriched token database.

WordNet tables used of this study are words table and senses table. Every token in the token table will be matched with each lemma in words table. Every lemma has one wordid and a wordid has several related lemmas. System will choose top lemma based on the tagcount order[2]. Selected lemma will be expanded into token table.



```
SELECT *
FROM `senses`
WHERE synsetid = '106626039'
LIMIT 0 , 30
```

Profiling [Inline] [Edit] [Explain

Show : Start row: 0 Number of rows: 30 Headers every 100 rows

Sort by key: None

+ Options

	wordid	casedwordid	synsetid	senseid	sensenum	lexid	tagcount	sensekey
Edit Copy Delete	48430	NULL	106626039	68004	1	1	38	film%1:10:01::
Edit Copy Delete	49457	NULL	106626039	69593	3	0	0	flick%1:10:00::
Edit Copy Delete	86937	NULL	106626039	120053	1	0	9	motion_picture%1:10:00::
Edit Copy Delete	86944	NULL	106626039	120052	1	0	0	motion-picture_show%1:10:00::
Edit Copy Delete	87257	NULL	106626039	120517	1	0	26	movie%1:10:00::
Edit Copy Delete	87276	NULL	106626039	120540	1	0	0	moving_picture%1:10:00::
Edit Copy Delete	87282	NULL	106626039	120536	1	0	0	moving-picture_show%1:10:00::
Edit Copy Delete	100096	NULL	106626039	137432	1	0	0	pic%1:10:00::
Edit Copy Delete	100205	NULL	106626039	137616	6	2	3	picture%1:10:02::
Edit Copy Delete	100216	NULL	106626039	137631	1	0	0	picture_show%1:10:00::

Figure 1. Senses Table of WordNet Database

Figure 1 shows how WordNet database works. For instance, movie lemma belongs to wordid 106626039. And wordid 106626039 has several lemmas like film, flick, motion picture, movie and many more. Evidently film lemma has the biggest tagcount and film lemma haven't listed on token table yet. Then film lemma will be added into token table so tokens of representative document become rich and vary.

D. TF-IDF Weighting Scheme

TF-IDF method is used to represent text in vector form [5]. TF-IDF method is described in formula (1), (2) and (3).

$$tf_{d,t} = f_{d,t} \tag{1}$$

$$idf_t = (\ log\ (N/df_t)+1\ ) \tag{2}$$

$$wf_{d,t} = tf_{d,t}\ .\ idf_t \tag{3}$$

Where :

$tf_{d,t}$ = *term frequency* , amount of token t in document d

$wf_{d,t}$ = weight value of document d

idf_t = *inverse document frequency*

N = total documents

df_t = amount of documents d where token t occurs

E. IB1 Algorithm

Instance-based Learning (IBL) is classified into lazy classifier for storing all of training instances and does not really work until the classification process runs. IB1 relatively accurate in classifying but require a large storage space for storing all the training instances. IBL algorithms classify instance by comparing it with training data (predefined class) so that a similar instance will have a similar classification using the cosine similarity function. Class selection is done by taking a number n of the highest similarity value class that became the most widely classification results. Each test document that correctly classified will be added in the training document. For the concept of IB1 algorithm described in Figure 2.

```
for each x ∈ Test Set do
     1. for each y ∈ Training Set do
        sim[y] ← Similarity(x,y)
     2. y_max ← some y ∈ Training Set with maximal Sim[y]
     3. if class(x) = class(y_max)
        then classification ← correct
        else classification ← incorrect
     4. Training Set ← Training Set ∪{x}
```

Figure 2. IB1 Algorithm. (Aha, Kibler., & Albert, 1991)

To determine the level of similarity, this study uses Cosine Similarity to calculate the distance between two points of the document and generate a list of matching training instance and test instance sorted by degree of

[1] WordNet's web reference (http://wordnetcode.princeton.edu/3.0/WordNet-3.0.tar.gz)

[2] Tagcount represents the decimal number of times the sense is tagged in various semantic concordance texts. A tagcount of 0 indicates that the sense has not been semantically tagged. (https://wordnet.princeton.edu/man/senseidx.5WN.html)

similarity. Cosine Similarity obtained from the equation (4).

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \tag{4}$$

Where :

A = test document
B = training document
n = amount of terms in a document
i = index of a term

IV. METHODOLOGY

A. Literature

The early stage of research is literature study of textbooks, journals and e-books that discuss about text mining, information retieval, sentiment analysis, data preprocessing, classifier algorithms, and evaluation methods.

B. Data Collection

Data set consists of 120 film reviews (2012 - 2015) from http://www.rottentomatoes.com. Data derived from the various movie genres like action, drama, comedy, science fiction and suspense. For each film, 10 reviews taken and they consist of 5 positive reviews and 5 negative reviews. 100 of the total documents will be used for training set and the rest 20 documents will be used as test set. Review taken from the web page and inserted into the database.

C. System Implementation

Each text document will be represented as vector. Elements of the vector is the weight of each term that appears in a document. The vector will be the object of IB1 algorithm. In classification process, output of the system is the class (positive or negative review) of the test document.

D. Evaluation

Evaluation of system performance can be measured by using evaluation methods like Precision, Recall and F-measure. Relevant documents used to evaluate system are test documents. Comparison class result and class from rottentomatoes.com are used to determine the relevance.

E. Analysis

The points of evaluation result analysis are the number of similar instances (k) and the use of WordNet.

V. EXPERIMENTS AND RESULTS

In the evaluation of this classification system, total documents are 120 documents in which 100 documents are used as training documents and 20 documents for testing. Results of the evaluation are value of tp, tn, fp, fn for all documents. Value of tp, tn, fp, fn are used to calculate the value of Recall, Precision and accuracy of the system. Evaluation is also based on the selection of instances and the use of WordNet. For evaluation based on the number of instances or training documents that have the highest similarity value, the instances selection are 5, 7 and 9 instances of the classification results that have been sorted by the highest similarity value. For WordNet evaluation is divided into evaluation using WordNet and evaluation without using WordNet.

TABLE I. CONFUSION MATRIX

	FACT		
SYSTEM		+	-
	+	tp	fp
	-	fn	tn

tp = true positive, number of correctly classified document to positive class

tn = true negative, number of correctly classified document to negative class

fp = false positive, number of negative labeled document that classified into positive class

fn = false negative, number of positive labeled document that classified into negative class

TABLE II. EVALUATION RESULT

Without Word Net											
k	tp	tn	fp	fn	Positive Class			Negative Class			A
					R	P	F	R	P	F	
1	6	6	4	4	0.6	0.6	0.6	0.6	0.6	0.6	60%
3	6	7	3	4	0.67	0.6	0.63	0.64	0.7	0.67	65%
5	8	8	2	2	0.8	0.8	0.8	0.8	0.8	0.8	80%
7	7	10	0	3	1	0.7	0.82	0.77	1	0.87	85%
9	10	9	0	1	0.91	1	0.95	1	0.9	0.95	95%
Using Word Net											
1	8	8	2	2	0.8	0.8	0.8	0.8	0.8	0.8	80%
3	9	9	1	1	0.9	0.9	0.9	0.9	0.9	0.9	90%
5	8	10	0	2	1	0.8	0.89	0.83	1	0.91	90%
7	7	8	2	3	0.78	0.7	0.74	0.73	0.8	0.76	75%
9	8	9	1	2	0.89	0.8	0.84	0.82	0.9	0.86	85%

Table 2 shows the evaluation results with and without using WordNet. Where k is the number of instances or nearest neighbor of classification results that sorted by highest similarity distance. Then R is Recall, P is Precision, F is F-Measure and A refers to accuracy.

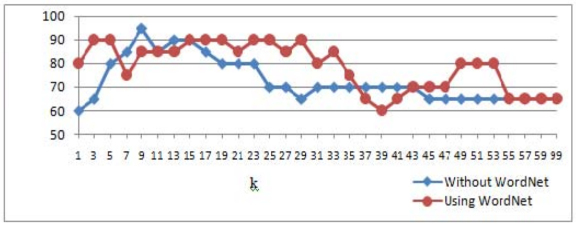


Figure 3. Graph of System Accuracy

Figure 3 shows the results of the accuracy using WordNet and accuracy without using WordNet in which the x axis shows the value of k and the y axis shows the percentage of obtained accuracy. Average accuracy without using WordNet is 72.7% while the average accuracy by using WordNet generates a value of 78.7%. Optimal WordNet occurs when instances set at the number 1 to 33 or k ≤ 30% of the amount of training documents. While the classification without WordNet optimal when k ≤ 20% of the amount of training documents. The objective of WordNet implementation is to expand the existing token by synonymous tokens so that the number of tokens became more widely practiced. From the results, the use of WordNet produce better results.

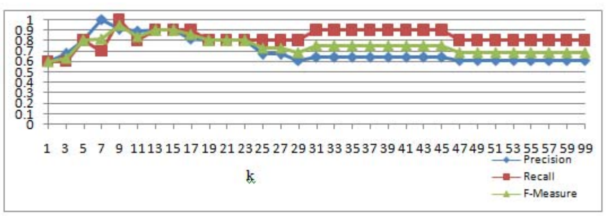


Figure 4. Graph of Positive Class Evaluation Without WordNet

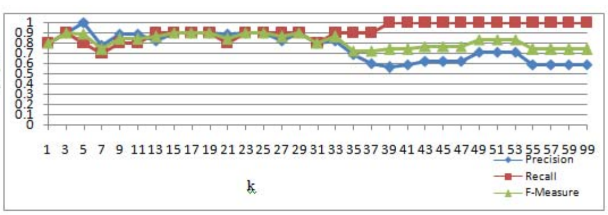


Figure 5. Graph of Positive Class Evaluation Using WordNet

Figure 4 and Figure 5 show the comparison of the value of Precision (P), Recall (R) and F-Measure (F) of positive category based on the presence or absence of the use of WordNet. The most optimal value for positive class without using WordNet appears when instances = 9 with Precision, Recall and F-Measure of 0.91, 1 and 0.95. By using WordNet, optimal values appeared when instances set at 3, 5, 15, 17, 19, 23, 25 and 29 with Precision, Recall and F-Measure value of 0.9.

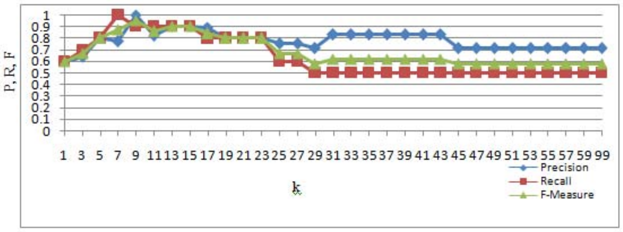


Figure 6. Graph of Negative Class Evaluation Without WordNet

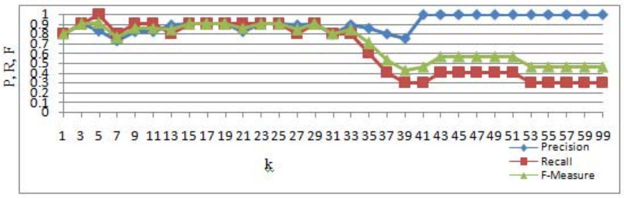


Figure 7. Graph of Negative Class Evaluation Using WordNet

Figure 6 and Figure 7 also show the comparison of the value of Precision (P), Recall (R) and F-Measure (F) of negative class based on using WordNet or without using WordNet. The most optimal value for the negative class without using WordNet appears when instances set on 9 with Precision, Recall and F-Measure of 1, 0.9 and 0.95. Use of WordNet reaches optimal value when instances on k = 5 with a Precision value of 0.83, Recall value of 1 and F-Measure value of 0.91.

From the results of evaluation that have been done to prove that the optimal k is small odd number. And 9 similar instances produce optimal results of the evaluation either without using WordNet or using WordNet, then 9 similar instances are used to compare the results of k-fold validation where representation documents are divided into six sections (k = 6) for a total of 120 documents. And every possibility of the document will be tested and seen the results of classification.

TABLE III. K-FOLD VALIDATION (WITHOUT WORDNET)

Iteration	Similar instances	k	Positive Class (+)			Negative Class (-)			Accuracy
			Precision	Recall	F Measure	Precision	Recall	F Measure	
1	9	6	0.91	1	0.95	1	0.9	0.95	95%
2	9	6	0,86	0,6	0,71	0,7	0,9	0,79	75%
3	9	6	0,63	1	0,77	1	0,4	0,57	70%
4	9	6	0,89	0,8	0,84	0,82	0,9	0,86	85%
5	9	6	0,64	0,7	0,67	0,67	0,6	0,63	65%
6	9	6	0,75	0,9	0,82	0,875	0,7	0,78	80%

Table 3 shows the results obtained from the k-fold validation with k (iteration) of 6 and Similar instances = 9. The average result of accuracy is 78.3% with the highest accuracy of 95% and the lowest accuracy by 65%. It shows the results of the classification remain high despite the testing documents and training records changed. It can be concluded that all the documents can be used to train and test documents without sacrificing value classification results.

TABLE IV. K-FOLD VALIDATION (USING WORDNET)

Iteration	Similar instances	k	Positive Class (+)			Negative Class (-)			Accuracy
			Precision	Recall	F Measure	Precision	Recall	F Measure	
1	9	6	0.89	0.8	0.84	0.82	0.9	0.86	85%
2	9	6	0,86	0,6	0,71	0,7	0,9	0,79	75%
3	9	6	0,77	1	0,87	1	0,7	0,82	85%
4	9	6	0,89	0,8	0,84	0,82	0,9	0,86	85%
5	9	6	0,7	0,7	0,7	0,7	0,7	0,7	70%
6	9	6	0,71	1	0,83	1	0,6	0,75	80%

The results of Table 4 obtained from the k-fold validation with k at 6 and Similar instances = 9. Average accuracy result is 80% with the highest accuracy by 85% and the lowest accuracy by 70%. It shows the results of the classification remain high despite the testing documents and training records changed. It can be concluded that all the documents can be used to train and test without hurting the classification results. In the comparison of results from table 3 and table 4 can also be concluded that average classification experiments using WordNet yield better evaluation results and better accuracy.

VI. CONCLUSION AND FUTURE STUDIES

Based on the analysis above, the use of WordNet gives effect in improving the accuracy of the system. WordNet is optimal when instance set at the number 1 to 33. From the results, the use of WordNet produce better accuracy (78.7%) than without using WordNet (72.7%). On the whole experiment, IB1 algorithm delivers performance above average at 65% to 95%. For further research can be used other tables in the WordNet database to optimize the use of WordNet, using store procedure to speed up the process of classification, implement stemming in preprocessing to minimize vector’s dimension and the identification of sentiment based on the context of sentences and phrases by applying methods of natural language processing.

REFERENCES

[1] D.Aha, D.Kibler and M.Albert, "Instance-Based Learning," 1991

[2] A.D.Cufoglu, " Classification Accuracy Performance of Naïve Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules(LBR) and Instance-Based Learner (IB1) – comparative study," Cairo: University of Westminster, June, 2009.

[3] R.Feldman, J.Sanger, "The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data," Cambridge: Cambridge University Press, 2007.

[4] J.Han, & M.Kamber, "Data Mining: Concepts and Techniques, 2nd Edition," San Fransisco: Morgan Kauffman Publishers, 2006.

[5] C. D.Manning, P.Raghavan and H.Schutze, "Introduction to Information Retrieval," New York: Cambridge University Press, 2008.

[6] B.Ohana, B.Tierney, "Sentiment classification of reviews using SentiWordNet," 9th. IT&T Conference, Dublin Institute of Technology, October, 2009.

[7] N. D.Sahoo, "The Impact of Online Product Reviews on Product Returns and Net Sales," 2013

[8] V. Soni, M. R. Patel, "Unsupervised Opinion Mining From Text Reviews Using SentiWordNet," IJCTT vol.11, pp. 234-238, May 2014.

[9] S.Vijayarani, M.Muthulakshmi, "Comparative Analysis of Bayes and Lazy Classification Algorithms," IJARCCE, 2013.