TELKOMNIKA, Vol.14, No.3, September 2016, pp. 1067~1076
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v14i3.2671

A Sentiment Knowledge Discovery Model in Twitter's TV Content Using Stochastic Gradient Descent Algorithm

Lira Ruhwinaningsih*, Taufik Djatna
Bogor Agricultural University, Bogor, Indonesia
Jl. Raya Darmaga, Kampus IPB Darmaga, Bogor 16680. Phone: +62 251 8622642
*Corresponding author, e-mail: purple_fim@yahoo.com

Abstract

Explosive usage of social media can be a rich source for data mining. Meanwhile, increasing and variation of TV shows or TV programs motivate people to write comments via social media. Social network contains abundant information which is unstructured, heterogeneous, high dimensional and a lot of noise. Abundant data can be a rich source of information but it is difficult to be manually identified. This research proposed an approach to perform preprocessing of unstructured, noisy and very diversed data; to find patterns of the information and knowledge of user activities on social media in form of positive and negative sentiment on Twitter TV content. Some methodologies and techniques were used to perform preprocessing. There were removing punctuation and symbols, removing number, replacing numbers into letters, translation of Alay words, removing stop word, and Stemming using Porter Algorithm. The methodology to used find patterns of information and knowledge of social media in this study is Stochastic Gradient Descent (SGD).The text preprocessing produced a more structured text, reduced noise and reduced the diversity of text. Thus, preprocessing affected to the accuracy and processing time. The experiment results showed that the use of SGD for discovery of the positive and negative sentiment tent to be faster for large and stream data. The Percentage of maximum accuracy to find sentiment patterns is 88%.

Keywords: *stochastic gradient descent, opinion mining, sentiment analysis, stemming porter, stream data mining*

1. Introduction

The variety of TV Program was presented to the audiences can be good quality content, touching, making people annoyed, monotonous and so forth. People may comment on the Twitter TV content. Twitter TV content is the twitter account that used by TV program providers as a media to comment on their TV shows, example: @Kikandyshow, @matanajwa and other.These comments can be useful information for TV content provider or become a rich source of information for data mining. Billions of information can be taken from social media web pages. Text comments are good indicators of user satisfaction. Sentiment analysis algorithms offer an analysis of the users' preferences, in which the comments may not be associated with an explicit rating. Thus, this analysis will also have an impact on the popularity of a given media show. Automatic TV content analysis is very important for the efficient indexing and retrieval of the vast amount of TV content available not only for end users so that they can easily access content of interest, but also for content producers and broadcasters in order to identify copyright violations and to filter or characterize undesirable content. The informations were captured from social media produce gold useful information for the organization. Mining thousands of viewing choices and millions of patterns, advertisers and TV networks identify household characteristics, tastes, and desire to create and deliver custom targeted advertising [1-3]. The background of this research is to provide information about the positive or negative sentiment to the TV viewer and content provider. Beside, TV content providers need to do improvement in every episode so that the content useful qualified and give a positive value to the society. TV content that taking a lot of audience needs continuous improvement to avoid monotony and lose audiences.The processed data is real time and stream data from twitter social media that every time can be

Received September 11, 2015; Revised May 10, 2016; Accepted May 24, 2016

known by the user. There is a number of quality attributes from the TV viewer view point there are: relevance, coverage, diversity, understandability, novelty, and serendipity [4].

The problem of this research consists of two. The first, social media data contain abundant information which is unstructured, heterogeneous, high dimensional and a lot of noise. Second, Abundant data can be a rich source of information, but it is difficult to identify manually and how to find patterns of information and knowledge of social media user activities in the form of positive and negative sentiment on twitter TV content. Vast, Noise, distributed and unstructured are characteristics of social media data [5]. Preprocessing can improve the accuracy and efficiency of mining algorithms involving distance measurements. Feature selection techniques is one of the most important and frequently used in preprocessing data mining. This technique reduces the number of features that are involved in determining a target class values, reducing the irrelevant feature, redundant, and the data that tend to misunderstanding of the target class that makes immediate effect for application. Social media data is a stream data that requires methodologies and algorithms can operate with a limited resource both in terms of time or memory hardware. Moreover, it can handle data that changes over time [6-8].

The proposed solution to the above problems is by making some modules for preprocessing and data mining modeling using SGD method. Tweet data was taken using Twitter Streaming API provided by twitter. The data is performed preprocessing which eliminates punctuation and symbols, eliminating number, replace numbers into letters, translation of Alay words and eliminate stop word. The next step is performing word stemming using Porter algorithm approach for Indonesia language. Preprocessing is an important step to the classification process and necessary for cleansing social media data that filled with noise and unstructured so that the data is ready to be processed to the next step. Preprocessing was made for Indonesia language and modules of preprocessing algorithm created by the author except stemming algorathm which adopted the Porter algorithm for Indonesa language. Classification algorithm was used to find patterns and information in this study is Stochastic Gradient Descent. SGD is suitable for large and stream data. Stochastic Gradient Descent is versatile techniques that have proven invaluable as a learning algorithm for large datasets. From the research that has been done, SGD model give effect to a short processing time to process a lot of data or data streams. Research about knowledge discovery on Twitter streaming data had been done by Bifet and Frank. Based on some tests that have been carried out, SGD models recommended for the data stream with the determination of the appropriate learning rate [8-9]. Previous research on the positive and negative sentiment has been done by Putranti and Winarko [10], namely twitter sentiment analysis for Indonesian language with the Maximum Entropy and Support Vector Machine. Research on sentiment analysis on Twitter data has been done using kernel tree and feature-based models [11]. Then, Taboada and his team have been doing research on Lexicon-Based Methods for extracting sentiment from text [12]. In addition, Opinion mining and sentiment analysis on a Twitter data stream based on their emotional content as positive, negative and irrelevant by Gokulakrishnan [13].

There were two main objectives of this study: to perform preprocessing to address unstructured data, a lot of noise and heterogeneous or diverse; find patterns of information and knowledge of social media user activities in the form of positive and negative sentiment on twitter TV content.

2. Research Method

The case study in this research was to determine the positive or negative sentiment based on tweet of TV content. The tweet data taken using Twitter Streaming API. Then it is taken continuously and stored in a table in real time. In the database, the tweets that have been collected at the table, processed (parsed), and its result distributed to several tables for preparation on the next process.

2.1. Preprocessing

To answer the problem of unstructured data, diverse and lots of noise some preprocessing methods and techniques carried out. The above are the techniques and methodologies were used in the preprocessing in this research. In general, the preprocessing can be described on Figure 1.

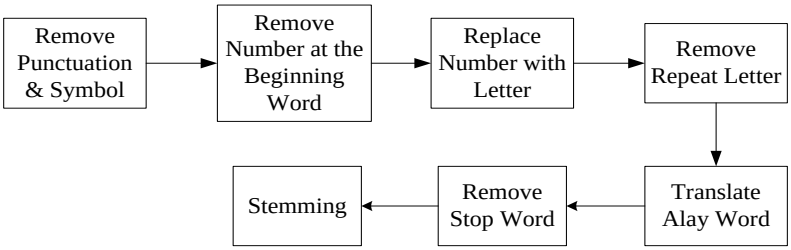

Figure 1. Preprocessing stages

2.1.1. Eliminate All Punctuation and Symbols

This information generally does not add to the understanding of text and will make it harder to parse the words on some comments on Twitter. This information includes single and double quotation marks, parentheses, punctuation, and other symbols such as dollar signs and stars. For examples, "*Haruskah kita membayar kembali negeri ini!!, tp selama kita terus bertanya apa negara ini dapat memberikan.*" Christine Hakim #MN, and after the process of removing punctuation and symbols, the sentence becomes "*Haruskah kita membayar kembali negeri ini, tp selama kita terus bertanya apa negara ini dapat memberikan.*" Christine Hakim #MN.

2.1.2. Eliminate Numbers in front of the Words

For examples, "*2hari*" becomes "*hari*". Just like remove symbol, remove number also use match character function to remove number by regular expression of .net library.

2.1.3. Replace Number with Letter

The following is an algorithm for replace number with letter:

1. Get list data conversion from database base on Table 1
2. Insert list data conversion to hash variable
3. Check numeric character in the middle of input string and check if it is not contained all numeric character using regex, if true then:
   Check wether it is contained string "00", if true then replace it by "u" else if it is contained string that match with data in Table 1, it will be replaced by convertion data from Table 1.
4. If point c is false, then return the origin of input string.

Table 1. List of converting the numbers into letters

Number	Conversion
0	o
00	u
1	i
2	Copy char before “2”
3	e
4	a
5	s
6	g
7	t
8	b
9	g

2.1.4. Eliminate Repeated letters

Examples: "*Pagiiiiiiiiii*" becomes "*pagi*". Check if it has repeated letter then take the first character only.

2.1.5. Translating the “*Alay*” Words into Normal Words

Alay word is excessive or strange words and also use writing mixed numbers at once or using uppercase and lowercase letters that are not reasonable [14]. It is stored in the database

and called as Alay data dictionary. The program will check into database whether the word included the Alay words or not, and then translate the words Alay into normal words. Examples: "*eM4Nk 4LaY GhHeToO*" becomes "*emang alay gitu*".

2.1.6. Eliminating the Stop Word

Stop word is a word that has no meaning and do not contribute during the processing of the data. Stop word is also stored in the database data as data stop words dictionary are like Alay data. If there are words that are listed in the table stop word, the word is removed. Examples: "*Jika dia pergi*" becomes "*pergi*".

2.1.5. Next Step Is Perform the Word Stemming

Stemming is the process of transformation of words contained in a document to basic word (root word) with certain rules. The algorithm that used to stemming is called Algorithm Porter for the Indonesia language [15]. The following is Porter algorithm:

Step 1 : Removing the particles.
Step 2 : Removing possessive pronoun.
Step 3 : Delete the first prefix. If there is a second prefix, go to step 4.1, if there is suffix then go to step 4.2.
Step 4 :
  Step 4.1 Removing second prefix, proceed to step 5.1.
  Step 4.2 Removing suffix, if it is not found then the word is assumed to be the root word. If found then proceed to step 5.2f
Step 5 :
  Step 5.1 Removing suffix, then the final word is assumed to be the root word.
  Step 5.2 Deleting a second prefix, then the final word is assumed to be the root word.

2.2. Classification Model Using Stochastinc Gradient Descent (SGD)

The reason of the use of SGD in this study because the data being processed is a data stream that is continuously updated each time. The stream data has characteristic that flows continuously, very large and real time. SGD has the ability to use only one training sample from training set to do the update for a parameter in a particular iteration. In the processing of stream data, the data do not wait until pilling up but the data processing is done continuously in real time as long as the data is being streamed. In most machine learning, the objective function is often the cumulative sum of the error over the training examples. But the size of the training examples set might be very large and hence computing the actual gradient would be computationally expensive. In SGD method, compute an estimate or approximation to the direction move only on this approximation. It is called as stochastic because the approximate direction that is computed at every step can be thought of a random variable of a stochastic process. There are two main parts in machine learning namely, training and testing. The data model generated from the training process is used for the prediction process in sentiment discovery. This is an answer to the second problems in this research.

First of all, the tweet data that have been labeled positive (yes) and negative (no) stored in Attribute-Relation File Format (ARFF) file. The reason for using the ARFF file because SGD library that is used comes from the Weka application. An ARFF file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files have two distinct sections. The first section is the Header information, which is followed the data information. The header of the ARFF file contains the name of the relation, a list of the attributes or the columns in the data. An example header on the dataset looks like this:

```
@relation 'MyArffTVContent'
@attribute text string
@attribute class {no,yes}
```

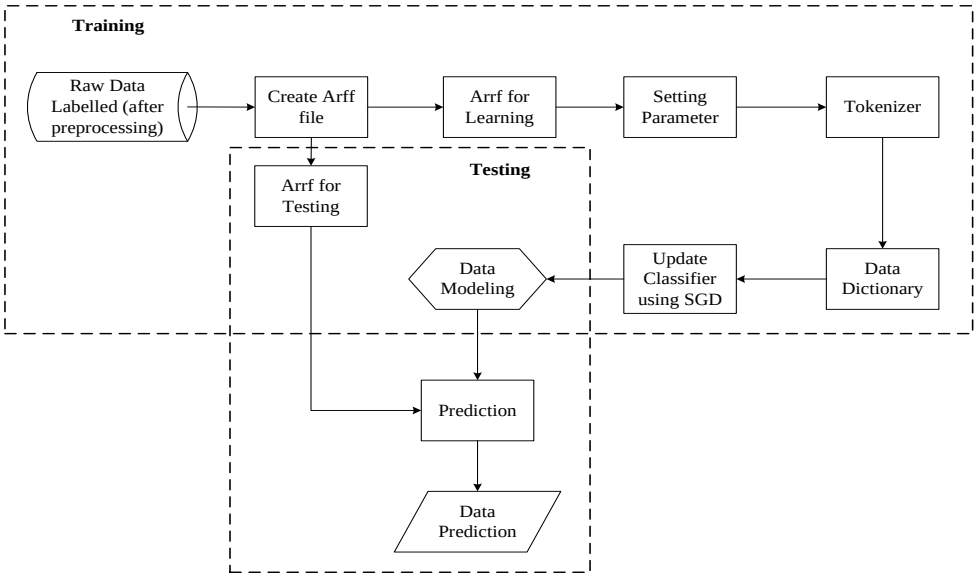

Figure 2. Flow classification model

The Data of the ARFF file looks like the following:

@data
'Semalam nangis KickAndyShow ',no
'kickandyshow eyuyant kartikarahmaeu lisaongt stargumilar lamat yg tang ',no
'KickAndyShow kau jodohkukenapakarna jodoh ngatur jodoh aturJombloMahGitu',yes
'KickAndyShow om saya suliat regist log',no

ARFF data is divided into two parts, one for training and another one for testing. The process flow is assumed for the evaluation model based on the presentation of training and testing. The determination of the value of a presentation is based on user input. There are some parameters in the training process, examples learning rate, epoch, lambda and etc. The next process is the tokenizer. In lexical analysis, tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens.

SGD is the methodology used to find patterns of information in the form of positive and negative sentiment on twitter TV content on this research. Stochastic Gradient Descent (SGD) is an algorithm that is used for large-scale learning problems. SGD has an excellent performance in solving large-scale problems. SGD efficient in performing classification even if it is based on non-differentiable loss function (*F(θ)*) [4].

$$F(\theta) = \frac{\lambda}{2}||w||^2 + \sum[1 - (yxw + b)]$$

w is the weight vector, b is the bias, $\boldsymbol{\lambda}$ is a regulation parameter, and the class label y assumed {+1, -1}. Explicit specification of learning rate is an important thing that relates to the time change in the data stream. Bifet and Frank have investigated research on the Twitter streams of data using several methods: Multinomial Naive Bayes, Stochastic Gradient Descent and Hoeffding Tree. Based on the tests that have been done of the three methods, models SGD with the appropriate learning rate is recommended to process the data stream. The formulation of update classifier in SGD is (F_{new}) [8]:

$$F_{new} = \sum_{i=0}^{n} \left[\sum_{i=0}^{n} w.\left(\frac{1-(\alpha.\lambda)}{n}\right) + \alpha\ .\ y \mid y(wx + b)| \right]$$

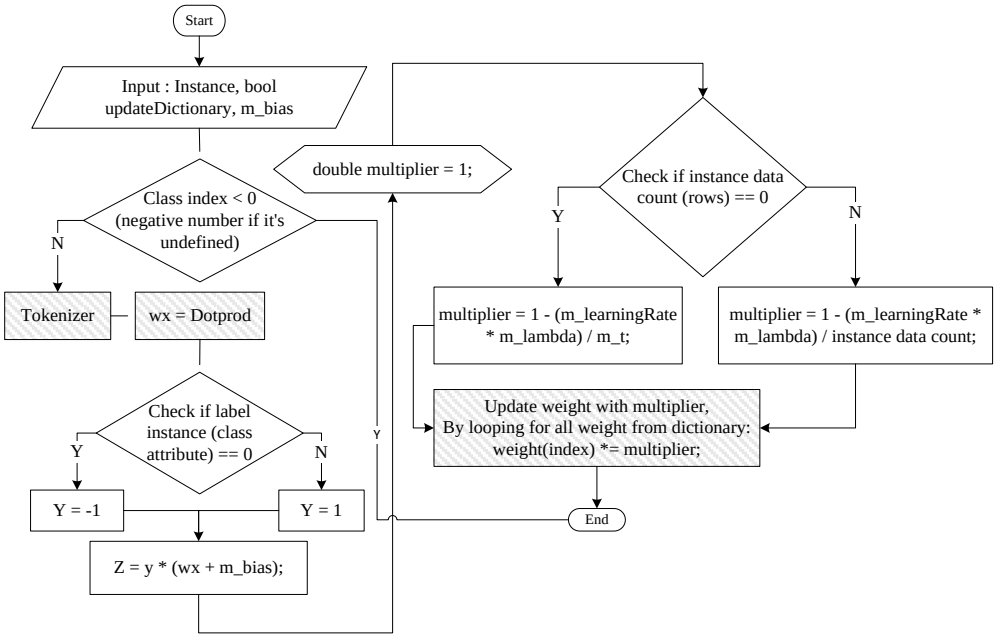

Figure 3. Flow update classifier part 1 referenced from Weka libraries

Where:
n = number of data tweet rows
i = sum of iteration
w = weight
α = Learning Rate
λ = Lambda
y = Status of sentiment (yes or no)
wx = accumulation of weight (per word)
b = bias.

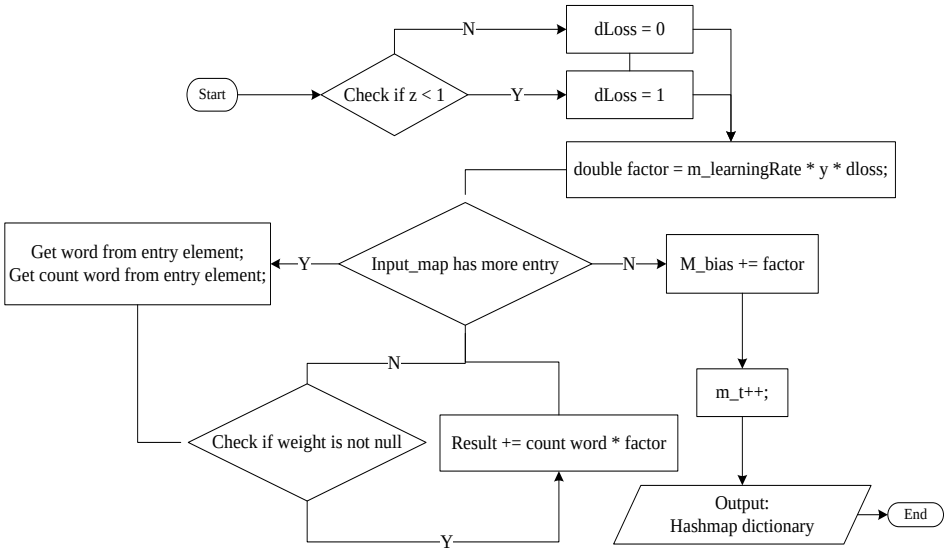

Figure 4. Flow update classifier part 2 referenced from Weka libraries

Instance data is stored in hashmap dictionary, which is a structure of data based on hashing, and allows storing the object as key value pair. In this case HashMap (key, object

value) means that each record data has a key word and an object value. Beside that, its object value is an object that contained two data values, which are number of word and weight. If data instance is greater than zero, then data raw position to be placed at random position. The goal is data input can be closer to the real condition in implementation with different inputs.

In the function Update classifier there are several sub-processes, namely "tokenizer" and "dotprod". Tokenizer (tokenization) has a main process for breaking a stream of text up into words, phrases, or other meaningful elements called tokens. If class index is greater than zero, then next process is tokenizer. Input data of tokenizer is an instance data, and in this case, an instance data is one tweet data, that represented in hashmap. Tokenizer has some checking processes to validate data, before going to its main process. First, it validates input instance data, to check if it is NULL or not, and second checking if its attribute type is a string data type.

3. Results and Analysis

The purpose of this experiment was to determine the accuracy of the SGD method by changing some parameters. Moreover, to know how to influence some differences preprocessing process against classification results in accuracy and processing time. Total data is used for testing and learning is 745 tweets. And here are some of the experiments were conducted.

3.1. Evaluation Model using Split Test

3.1.1. Preprocessing on the Classification Accuracy and Processing Time

The first test is the analysis of preprocessing on the classification accuracy and processing time. Data is divided into two parts, 90% training data and 10% testing data. The graph below shows the variation of the preprocessing of the classification accuracy which tend to rise, when the data has through several stages of preprocessing. Correctly classified instance is accuracy in classifying the data. Significant increases occurred for data that has been through preprocessing step; those are remove stop word and the stemming. Step Remove stop word means to eliminate words that include a Stop word in the data tweet, because it does not contribute to process of classification. While the stemming will make words of tweet only consist the root word. The suffix, prefix particle, suffix will be eliminated in the process of stemming. The time required process tends to decrease if the data has been through in several preprocessing steps, shown in graph as illustrated in Figure 5.

Figure 5. Chart execution time on preprocessing variations

The next testing used learning rate = 0.01, epoch = 100, with variations of learning data, and also data testing. Data testing 90% until 10% using decrement step 10% and Data training 10% until 90%, and using increment step 10%. The data is divided into training and testing data. In the first experiment, the training data are set by 10% of the total data and data testing by 90% of the total data. Further testing by adding the data training and reduce the data testing. Figure

6 shows that the more training data, the classification accuracy will tend to rise. More training data will require increasing the execution time; it is shown in Figure 7.

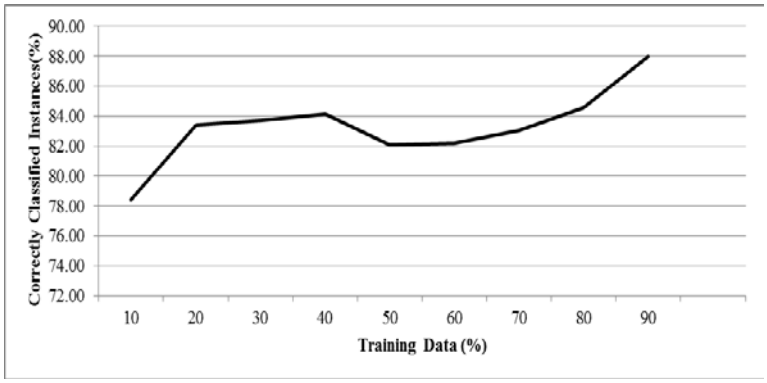

Figure 6. Correctly classified instances chart

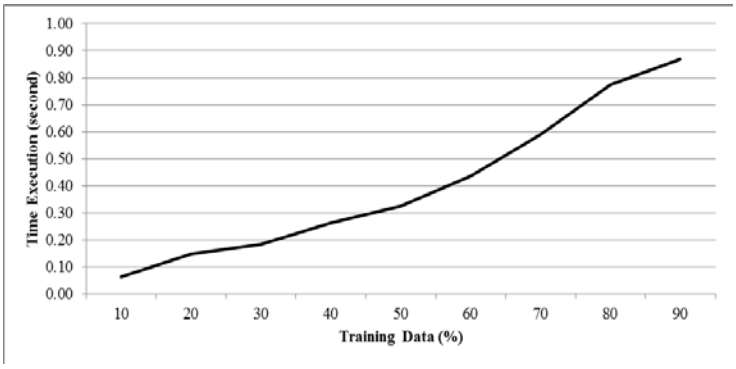

Figure 7. Total time execution Chart

3.1.2. Changing Learning Rate

The next experiment is changing learning rate, with epoch 100 for each variation of the amount of training and testing. Figure 8 shows the variations on learning rate and the amount of training data. Learning rate 0.001 gives the classification accuracy values which tend to rise. However, the effect of changing learning rate against the execution time is not too significant. In addition that illustrated in Figure 9, the learning rate is getting smaller, so the process will take a long time tend execution when the number of training data increases.

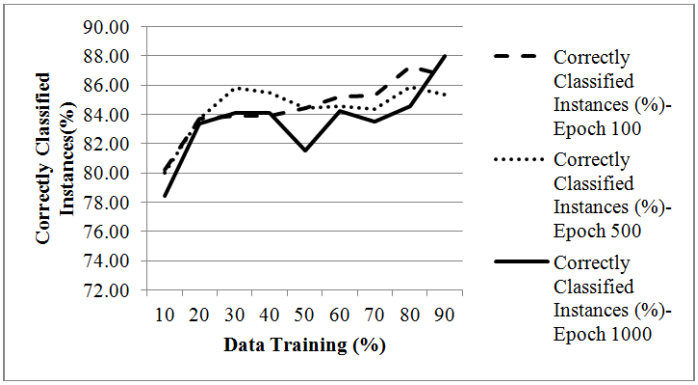

Figure 8. Chart of epoch changes to correctly classified instances

3.1.3. Epoch Changes to Correctly Classified Instances

Figure 8 shows that correctly classified Instances is highest during high training set and high epoch.

3.2. Evaluation Model using Cross validation

Testing by using cross validation with folds 2, 3, 4, 5, 6, 7, 8, 9, and 10 obtained correctly classified instances tend to be stable at around 84% it is seen in the following chart

Figure 9 shows measurement of correctly classified instance using cross validation with variations of folds number. Correctly classified instance looks to be around 84% whatever the number of folds.

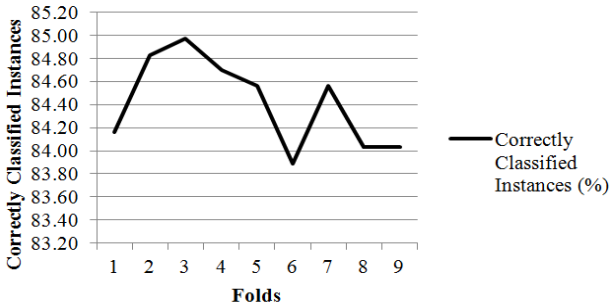

Figure 9. Chart correctly classified instances using Cross Validation

From the Figure 10 below shows that there are two classes are concentrated in two places. Top right is a class "Yes" the positive sentiment. While on the bottom left is a collection of class "No" is negative sentiment. In Figure 10 shows that sample pattern output captured that describe how sentiment point distribution for each word is placed. Red dot represented for positive sentiment words, and otherwise, blue dot describe for negative sentiment word. If its position is closer to tip, so it is signifies that sentiment is getting stronger and it makes its accuracy becomes high. As the description of findings and computation above, the advantages of this study revealed the strength of information hiding from TV audience's Twitters. Our preprocessing approach has succeeded to dicovered information from unstructured and noisy data. As so far, the whole processing time tended to shorter than previous studies about sentiment analysis on the stream data using Multinomial Naive Bayes and Hoeffding Tree. Some disadvantages include current problem in accuracy as well as the partial representation of how the evaluation of public media such as TV content scientifically computed. Based on the results of previous studies [3] accuracy Multinomial Naive Bayes methods are better than SGD.

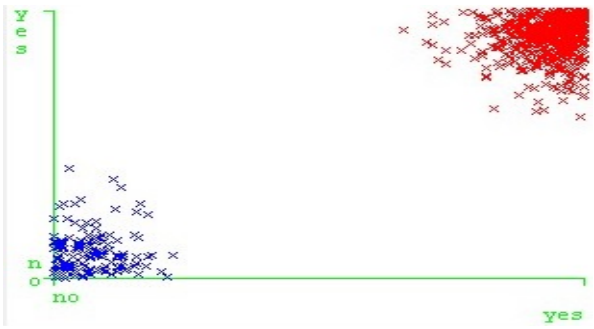

Figure 10. Sentiment pattern

4. Conclusion

The data used in this study is derived from the data twitter TV content such as comments or user tweet. Twitter account which became the object of this study is

@kickandyshow. Tweet data that has been through preprocessing can be more structured text, reducing noise and reducing the diversity. Preprocessing accuracy in eliminating all punctuation and symbols, eliminating numbers, replace numbers with letters, eliminating repeated letter reaches 100 % with the execution time of each approximately 2 milliseconds for 760 tweets. Whereas, the execution time for translating the stop word, eliminating stop words and stemming is 77, 111 and 1230 milliseconds respectively. Preprocessing stages affect to the higher accuracy of the classification and reduce processing time. This study successfully extracted patterns of information and knowledge of social media user activities in the form of positive and negative sentiment on twitter TV content. The results of the experiment showed that large amount of training data can affect to the accuracy of the classification of positive and negative sentiment but require a longer time for the training process. Learning rate changes little effect on duration of the process and the accuracy of classification. Learning rate is getting smaller than the execution time can increase by using large training data. Percentage of correctly classified instance is with a maximum of 88%.

In order to fulfill an ideal study, in the future it is recommended to search for a higher accuracy leveled algorithm that cover the whole process in TV content evaluation scientifically.

References

[1] Peleja F, Dias P, Martins F. A recommender system for the TV on the web: integrating unrated reviews and movie ratings. *Multimedia Systems*. 2013; 19(6): 1-16.

[2] Han J, Kamber M, Pei J. Data Mining Concepts and Techniques. Second Edition. San Francisco: Morgan Kaufman. 2012.

[3] Spangler WE, Gal-Or M, May JH. Using Data Mining to Profile TV Viewers. *Communication of the ACM*. Pittsburgh. 2003; 46(12): 67-72.

[4] Bambini R, Cremonasi P, Turiin R. TV Content Analysis: Recomender System for Interactive TV. Boca Raton: CRC Press. 2012.

[5] Gundecha P, Huan L. Mining Social Media: A Brief Introduction. In: Smith JC, Greenberg HJ. *Editors.* Tutorials in Operations Research: New Directions in Informatics, Optimization, Logistics, and Production. Eighth Edition. Arizona: INFORMS; 2012: 1-17.

[6] Han J, Kamber M, Pei J. Data Mining Concepts and Techniques. Second Edition. San Francisco: Morgan Kaufman. 2012.

[7] Djatna T, Yasuhiko M. Pembandingan stabilitas algoritma seleksi fitur menggunakan transformasi ranking normal. *Jurnal Ilmu Komputer*. 2008; 6(2): 1-6.

[8] Bifet A, Frank E. *Sentiment Knowledge Discovery in Twitter Streaming Data*. Proceedings of the 13th International Conference, DS 2010. Canberra. 2010; 6332: 1-15.

[9] Bottou L. Stochastic Gradient Descent Trick. In: Montavon G, Orr G, Muller K. *Editors*. Neural Network: Trick of the Trade. Second Edition. Heidelberg: Springer Berlin Heidelberg; 2012: 421-436.

[10] Putranti N, Winarko E. 2014. Analisis Sentimen Twitter untuk Teks Berbahasa Indonesia dengan Maximum Entropy dan Support Vector Machine. *JCCS*. 8: 91-100.

[11] Agrawal A, Xie B, Vovsha I, Rambow O, Passonneau R. *Sentiment Analysis For Twitter Data*. LSM '11 Proceedings of the Workshop on Languages in Social Media. Stroudsburg. 2011: 30-38.

[12] Taboada M, Brooke J, Tofiloski M, Voll K, Stede M. Lexicon-Based Methods for Sentiment Analysis. *MIT Press Journal*. 2011; 37(2): 267-307.

[13] Gokulakrishnan B, Priyanthan P, Ragavan T, Prasath N, Perera A. *Opinion Mining and Sentiment Analysis on a Twitter Data Stream*. IEEE Advances in ICT for Emergening Regions (ICTer) International Conference. 2012; 10: 182-188.

[14] Meyke. Penggunaan Kosa Kata Alay Oleh Remaja Pada Facebook di Kota Bengkulu. Master Thesis. Bengkulu: Postgraduate Bengkulu University; 2013.

[15] Agusta L. *Comparison of Porter Stemming Algorithm and Nazief & Adriani's Algorithm for Stemming Indonesian Text Documents*. National Conference and Information Systems 2009. Bali. 2009; 036: 196-201.