

Sentiment analysis enhancement with target variable in Kumar's Algorithm

A A Arman*, A B Kawi and R Hurriyati

Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jl. Ganesha 10 Bandung, Indonesia

*Corresponding author: arry.arman@yahoo.com

Abstract. Sentiment analysis (also known as opinion mining) refers to the use of text analysis and computational linguistics to identify and extract subjective information in source materials. Sentiment analysis is widely applied to reviews discussion that is being talked in social media for many purposes, ranging from marketing, customer service, or public opinion of public policy. One of the popular algorithm for Sentiment Analysis implementation is Kumar algorithm that developed by Kumar and Sebastian. Kumar algorithm can identify the sentiment score of the statement, sentence or tweet, but cannot determine the relationship of the object or target related to the sentiment being analysed. This research proposed solution for that challenge by adding additional component that represent object or target to the existing algorithm (Kumar algorithm). The result of this research is a modified algorithm that can give sentiment score based on a given object or target.

1. Introduction

Information has a very important role at this time. Information used primarily as a support in the decision making or simply just to add insight. At the beginning of Data Warehouse and Business Intelligence era, the data sources for analytical mostly come from internal. The data sources usually come from operational databases that spread in many databases in many business units across the organization. By this system, manager can decide something by doing cross analytical between databases from different business unit or different application, i.e. analysing relationship between selling and amount of marketing budget.

In other side, the concept of competitive intelligence start to implement widely. Organization start to collect or capture strategic information from the outside and combine with internal data sources to decide strategic decision. At the beginning, external information can be collected by conventional way. Today, when more and more business going online, when more and more people talk through social media, collecting information must conducted by different way.

Sentiment analysis (also known as opinion mining) refers to the use of text analysis and computational linguistics to identify and extract subjective information in source materials. Sentiment analysis is widely applied to reviews discussion that being talked in social media for many purposes, ranging from marketing, customer service, or public opinion of public policy. The result of sentiment analysis can be used as a source from external side.

There is a lot of sentiment research has been done. Sentiment analysis research used many different approaches such as corpora based approach or statistical-based approaches (Naive Bayes, EM, SVM).



Many of these studies were based on foreign language (mostly for English) that require adaptation in other languages. One of the popular algorithm for Sentiment Analysis implementation is Kumar algorithm that developed by Kumar and Sebastian [1]. Kumar algorithm can identify the sentiment score of the statement, sentence or tweet, but cannot determine the relationship of the object or target related to the sentiment being analysed. This research will try to propose solution for that challenge by adding additional component that represent object or target to the existing algorithm (Kumar algorithm). By this modification, modified algorithm can give new score that represent sentiment that already corrected by the factor of relationship to the object or target being analyzed.

2. Related Work

Research conducted by Go and friend [2] uses emoticons for classifying sentiment in English which previously inspired by research conducted by the Read [3]. Their research using data taken from the Twitter API as training data for their systems. The weakness of this work is they only divide sentiment polarities into 2 types positive and negative without neutral sentiment. Later research conducted by Mr. and Paroubek [4] who classify sentiment to the neutral class by using tweets coming from the official account of mass media as training data for neutral sentiment.

Different studies also conducted by Xue Chen [5] adapting algorithm that has long been used to topic modeling. Sentiment classification is done using DLDA algorithm. This algorithm is inspired by the first LDA algorithm that intended to classify documents into certain topics [6]. They use this algorithm because the underlying assumption is topic is a collection of words (bag of words).

Other research conducted by Chenliang Li [7] propose an algorithm for segmentation that called HybridSeg. Tweet segmented into sections that have meaning or context information and then it can be easily used by other applications. Li Chenliang get good accuracy results when used in NER (Named Entity Recognition).

The research that use data from Indonesian language is by Wicaksono, et al [8]. It used lexicon approach in building their corpora. The study showed an increase in performance with the use of lexicon. Research conducted by Kumar and Sebastian [1] using syntax approach applying to the “bag of words”. The research conducted by Jiang [9] who classify with target approach show improved performance compared to the classification solely on features of independent targets.

The majority of existing research is not considering object as the main feature and only focusing in the value of the sentiment. For example, “happy happy happy” making it as a positive sentiment but we have no idea to whom the text is intended.

Refer to sentiment definition from *cambridge.org*, sentiment is “a thought, opinion or idea that is based on the situation or way of thinking from something”. Objects become important and influential because of it is the purpose of sentiment. Target recognition in the text will determine the subjectivity of tweet. Definition of sentiment by Liu [10]. Liu said that the target is part of the sentiment that can be seen in this relation ($ei, aij, sijkl, hk, mp$) or *entity, aspect, sentiment, holder, time*.

Based on these, this research use the object as one of its features in sentiment classification. This Research will design, develop, and implement a modified algorithm to calculate sentiment related to particular object. The algorithm is an enhancement from an algorithm developed by Kumar and Sebastian. We use this algorithm based on the advantage from its simple computational level that can quickly deliver results, and good enough to rely on with the vocabulary provided. All these advantages are not complete without the ability of the algorithm to handle the targets of sentiment. Target of sentiment is very important when the algorithm is applied to the real world or used in the specific environment.

3. Understanding Kumar Algorithm [1]

Kumar algorithm is a sentiment classification algorithm developed by Akshay Kumar and Teeja Mary Sebastian with the main objective is to get the semantic orientation of opinion words in tweets [1]. Kumar algorithm distribute sentence or tweets into positive, neutral, or negative categories. This

algorithm is using a hybrid approach to utilize the advantages from both corpus and dictionary. This algorithm also uses the Machine Learning and Natural Language Processing techniques.

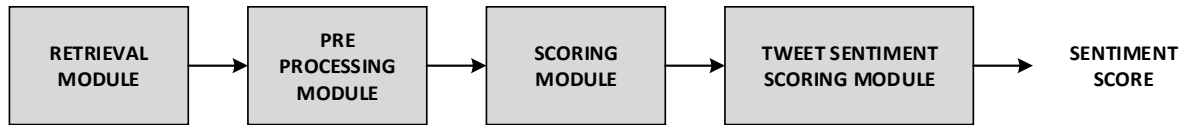


Figure 1. System Architecture of Sentiment Analysis System (modified from [1])

Classification is conducted by taking into account the emergence of words and characters that are considered to have roles in determining sentiment of tweets such as adjective, verb, and adverb. The calculations were carried out to map tweets into the range of -1 to 1. The advantages of these algorithms is the simplicity of its calculation as shown in equation 1. The weaknesses of this algorithm is that its performance depends on the completeness of the vocabulary.

$$S(T) = \frac{1 + \frac{Pc + \log(Ns) + \log(Nx)}{3}}{|OI(R)|} * \sum_{i=1}^{|OI(R)|} (S(AG_i) + S(VG_i) + N_{ei} * S(E_i)) \quad \text{Eq (1)}$$

Where,

$ OI(R) $	denotes the size of the set of opinion groups and emoticons extracted from the tweet,
Pc	denotes fraction of tweet in caps,
Ns	denotes the count of repeated letters,
Nx	denotes the count of exclamation marks,
$S(AG_i)$	denotes score of the i-th adjective group,
$S(VG_i)$	denotes the score of the i-th verb group,
$S(E_i)$	denotes the score of the i-th emoticon
N_{ei}	denotes the count of the i-th emoticon

The classification process is conducted by looking at the pieces of words contained in the tweet after pre-processing. In scoring module, sentence process to determine the value of the adjective group and adverb group. Value of adjective group is a combination of the occurrence of the adjective word with the adverb word. Its value is calculated by the following equation.

$$AG = nAdj * nAdv \quad \text{Eq (2)}$$

Where:

AG denotes value of Adjective Group.
 $nAdj$ denotes Adjective value.
 $nAdv$ denotes Adverb value.

The calculation of the value of the verb group which is the combination of the occurrence of the verb word and adverb is done by the following equation.

$$VG = nVb * nAdv \quad \text{Eq (3)}$$

Where:

VG denotes Value of Verb Group.
 nVb denotes Verb value.
 $nAdv$ denotes Adverb value.

Table 1. Table of Verbs and Adverbs

Verb	Score	Adverb	Score
cinta	1	sempurna, tuntas, sembuh, sehat	1
kagum	0.9	paling, amat	0.9
suka	0.8	sangat	0.8
nikmat	0.7	selalu	0.7
senyum	0.6	sekali	0.6
terkesan	0.5	banyak	0.4
tertarik	0.4	cukup	0.3
senang	0.3	lebih	0.2
santai	0.2	beberapa	-0.2
bosan, tolak	-0.2	agak	-0.3
jijik	-0.3	sedikit	-0.4
derita	-0.4	kurang	-0.6
tidak suka	-0.6	jarang	-0.8
sebal	-0.9	tidak	-0.9
benci	-1	tidak pernah	-1

Table 2. Table of Emoticons

Emoticon	Meaning	Score
:D	Laugh out loud	1
BD	Laugh out loud with glasses	1
XD, :)), :), =D>	Laugh	1
\m/	Hai5	1
:), =), :-), :3	Smile	0,5
.*	Kiss	0,5
:	Poker face	0
:\	Not decided	0
:(Sad	-0,5
>/3	Broken heart	-0,5
B(Sad with glasses	-0,5
:(Crying	-1
X-(Angry	-1

Table 3. Table of Adjectives

Adjective	Score
bahagia, ceria, gemar, gembira	0,5
babil, awawarna, awamineral, azal, baki, bangkas, batangan	0
agresif, akut, alergis, ambekan, aneh, amburadul, ceroboh	-0,5

The second calculation of the above equation is conducted by keeping in mind that if the pair not found, its value will be multiplied by 0.5 and the value of each word is taken based on research conducted by Kumar and Sebastian. If not found it will look for synonyms of words in Indonesian. The value of emoticons are also taken from Kumar and Sebastian's research with additions and adjustments as needed. The value of adjective's word is done by looking at the meaning of the word. Adjective words that have positive meanings will be given a value of 0.5, -0.5 for negative, and 0 for neutral.

Research conducted by Go and friend [2] uses emoticons for classifying sentiment in English which previously inspired by research conducted by the Read [3].

4. Proposed Algorithm

The proposed modification of Kumar algorithm only concern in the Tweet Sentiment Scoring Module. Again, the original algorithm has no parameter related to the object or target being analyzed. In order to add object parameter, we need to insert additional parameter that represent object, say it called $RS(Obj)$. After addition, the modified equation will be transform into new form as can be seen in Equation 4 and Equation 5.

$$SM(T, Obj) = S(T) * RS(Obj) \quad \text{Eq (4)}$$

$$SM(T, Obj) = \frac{1 + \frac{Pc + \log(Ns) + \log(Nx)}{3}}{|OI(R)|} * \sum_{i=1}^{|OI(R)|} S(AG_i) + S(VG_i) + N_{ei} * S(E_i) * RS(Obj) \quad \text{Eq (5)}$$

Where

- $SM(T, Obj)$ denotes the modified sentiment score of modified algorithm for Text T and Object Obj
- $S(T)$ denotes the sentiment score of Kumar algorithm
- $RS(Obj)$ denotes relationship score of Obj in the sentence
 RS will be zero if Obj not exist in the sentence
 RS will have positive value and less than 1

There is several requirements to define the $S(Obj)$. First requirement, $S(Obj)$ value should be always positive to maintain the polarity of previous sentiment result. Second requirement, the value should be zero when the object is not found. Its mean that no related sentiment to that object. In this case, $S(T)$ can have some value, but the sentiment value is not related to the object.

$$RS(Obj) = \begin{cases} 0 & \text{if } Obj \text{ not exist in } T \\ 1 - \frac{(i-1)}{N} & ; \text{ where } i = \text{word position of } Obj \text{ in } T; N = \text{number of word in } T \end{cases}$$

The second equation of RS will generate value correspond to the position of Obj Word in sentence T . Assume that position around the beginning is more important than the position around the end. First extreme condition, the Obj Word exist at the first position, $i=1$, $RS(Obj)=1$. Second extreme condition, the Obj Word exist at the last position (i.e $N=8$), $i=7$, $RS(Obj) = 1-7/8=1/8$. Value of RS will be maximum ($RS=1$) if Obj Word exist in the first position. The value will decrease if the position of Obj Word moving to the next position up to the end. At the end, the value become close to zero, but never reach zero. The value zero will be assign by the first equation (or rule) if Obj Word not exist in the sentence T . $SM(T, Obj)$ score will be same with the $S(T)$ for the case that Obj Word exist in the first position.

To compute the relationship score (RS), the system must scan the sentence T to find the position of Obj word. If it is found, it will be the value of i , and RS can be compute by the second equation of RS above. If Obj word not found, RS will be assigned as zero.

Obj Word can be given manually when user want to know sentiment score for any **Object** (represent by Obj Word). The other alternative, additional process can be apply in pre-processing module. There

is mechanism to identify all *Obj Word* in each sentence T . Finally, in Tweet Sentiment Scoring Module the computation of $S(T)$ and $SM(T, Obj)$ will be repeated according to the number of *Obj Word*. The result will be $SM(T, Obj)$ value for each *Obj Word*.

5. Evaluation

System evaluation will be conducted in the follows steps.

- 1) Collect sample of data to be classified from Twitter (1751 tweet samples).
- 2) Conduct “Manual Sentiment Analysis” from collected samples to categorize each sample into neutral, positive, or negative sentiment (will be used as a reference).
- 3) Preparing *Obj Word*
 - a) Identify sentence that have no object, and assign *Obj Word* as any NOUN, i.e. “people”. Remember, this type of sentence have no NOUN.
 - b) Identify all sentence that have object, choose one object (if have more than one object), and assign *Obj Word* with a correct object that available in that sentence.
 - c) Change several *Obj Word* in step (b) to the other Noun that not available in that sentence. The aim of this step to verify if the modified algorithm can identify *Obj Word* that not exist in the sentence.
- 4) Compute $S(T)$ and $SM(T, Obj)$ for each sentence T and *Obj*.

Table 4. Experiment Result

Analytical Methods	Neutral (0)	Positive (+)	Negative (-)	Change (+) to (0)	Change (-) to (0)	Change (+) to (-) or (-) to (+)
Manual	1026	549	176	-	-	-
Kumar Algorithm ($S(T)$)	865	676	210	-	-	-
Modified Algorithm ($RS(T, Obj)$)	897	653	201	23	9	0

Table 5 show the result of the experiment. Manual Sentiment Analysis of 1751 sentences identify that 1026 sentences is neutral, 549 sentences is positive, and 176 is negative. Identification of sentiment category by Kumar algorithm show that 865 sentence is neutral, 676 is positive, and 210 is negative. These result show that Kumar algorithm have accuracy less than 100%. But, the percentage is not important, because this research not to measure Kumar algorithm accuracy.

Put attention to the result of Modified Algorithm that compute $RS(T, Obj)$. The number of neutral sentiment is increased, and the number of positive sentiment and negative sentiment is decreased. Its happen because some sentence that previously in the category of positive (23) and negative (9) category was changed. In our experiment several sentence have no NOUN, so it will be change to neutral when it compute by modified algorithm, because any NOUN as an object will not found in the sentence. For other case, several sentence give the wrong object, so the *Obj Word* will be not found and give the result as neutral. In other hand, there is no sentence that change the polarity, because the algorithm guarantee that it will not happen. Changing from neutral to any polar also will not happen, because the $RS(Obj)$ maximum value is 1, and always positive. So the original value from Kumar algorithm will never increase.

6. Conclusion

Based on the evaluation result that already discuss in part 5, it can be concluded that in this paper already propose the modification of Kumar Algorithm. The modified algorithm add additional component to the

original algorithm. This additional component, $RS(Obj)$, represent the relationship of the object (Obj) being analysed. The list characteristics of the modified algorithm describe in the following lists.

- 1) The value range of RS is 0 to 1.
- 2) The value of RS will be zero if *Obj Word* that represent Object (or target) was not found in the sentence being analysed
- 3) The value of RS will be maximum (value=1) if the *Obj Word* found in the first position in the sentence.
- 4) The value of RS will be decreased if the position of *Obj Word* moving to the end of the sentence. At the end position, the value will close to zero, but never zero.
- 5) The final value of new algorithm ($SM(T, Obj)$) never bigger than the value from original Kumar algorithm ($S(T)$).

References

- [1] A. Kumar and T. M. Sebastian, "Sentiment analysis on twitter," *IJCSI International Journal of Computer Science Issues*, 2012.
- [2] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," *Processing*, vol. 150, pp. 1–6, 2009. [Online]. Available: <http://www.stanford.edu/~alecmgo/papers/TwitterDistantSupervision09.pdf>
- [3] J. Read, "Using Emoticons to reduce Dependency in Machine Learning Techniques for Sentiment Classification," *ACL Student Research workshop*, no. June, pp. 43–48, 2005.
- [4] A. Pak and P. Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." *Lrec*, pp. 1320–1326, 2010. [Online]. Available: [http://incc-tps.googlecode.com/svn/trunk/TPFinal/bibliografia/PakandParoubek\(2010\).TwitterasaCorpusforSentimentAnalysisandOpinionMining.pdf](http://incc-tps.googlecode.com/svn/trunk/TPFinal/bibliografia/PakandParoubek(2010).TwitterasaCorpusforSentimentAnalysisandOpinionMining.pdf)
- [5] X. Chen, W. Tang, H. Xu, and X. Hu, "Double lda: A sentiment analysis model based on topic model," in *Semantics, Knowledge and Grids (SKG), 2014 10th International Conference on*, Aug 2014, pp. 49–56.
- [6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2012.
- [7] C. Li, A. Sun, J. Weng, and Q. He, "Tweet Segmentation and its Application to Named Entity Recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 4347, no. 2, pp. 1–1, 2014. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6823714>
- [8] A. F. Wicaksono, C. Vania, B. D. T, and M. Adriani, "Automatically Building a Corpus for Sentiment Analysis on Indonesian Tweets," 2014, pp. 185–194.
- [9] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent Twitter Sentiment Classification," *Computational Linguistics*, pp. 151–160, 2011. [Online]. Available: <http://www.aclweb.org/anthology/P11-1016>
- [10] B. Liu, *Sentiment Analysis and Opinion Mining*, 2012, vol. 5, no. May.