International Journal of Advances in Computer Science & Its Applications – IJCSIA
Volume 4: Issue 4 [ISSN 2250-3765]
Publication Date : 27 December,2014

ASPIRE: Building a Sentiment Lexicon from Ratings of Social Reviews

Hamidreza Keshavarz-Mohammadian, Mohammad Saniee Abadeh

***Abstract*— Finding semantic orientation and intensity of sentiment phrases and words is a substantial task of opinion mining. The problem is to give a score to each sentiment phrase, so that different expressions of opinions in different platforms, like social networks, can be processed. There have been several attempts to do this task, and this paper aims to score each sentiment phrase based on its occurrence in reviews with different overall ratings. The idea is that if a sentiment phrase occurs more in 5-starred reviews than in 3-starred ones, it should be more positive and more intense. The results support this idea. Each sentiment phrase in the corpus is given a score based on a weighted average of their frequency in reviews with different ratings. When a sentiment phrase gets a high score, it means it is more likely to be positive and more likely to be intense. And if a sentiment phrase gets a low score, it means that it is negative. This score sets the threshold of negativity and positivity. The high precision and recall for this feature shows its significance in classifying positive and negative sentiment phrases.**

***Keywords*— Opinion Mining; Sentiment Analysis; Sentiment Lexicon Generation; Word Polarity; Sentiment Words**

I. Introduction

Opinion mining has gained a momentum in recent years; mostly because of the need to search online, to find out how good or bad a particular object is, and the sheer volume of available reviews in websites, blogs, discussion forums, social networks, and other various outlets. People nowadays share their experiences and views on social networks and review sites, and opinion mining helps in processing the opinions people express on different entities.

Opinion mining deals with analyzing people's opinion they express about entities and their attributes [1]. When someone wants to know about other people's sentiments towards a particular product, object, human, organization, or topic, they turn to Internet to search reviews.

The emergence of Web 2.0 enabled people to express their opinion about everything in every possible way on different websites, mostly on social media. But the huge amount of reviews written makes it virtually impossible to read every single review. Here comes opinion mining to process data, and to extract and summarize opinions.

Hamidreza Keshavarz-Mohammadian, Mohammad Saniee Abadeh
Faculty of Electrical and Computer Engineering, Tarbiat Modares University
Tehran, Iran

One of the most important tasks of opinion mining is sentiment lexicon generation [1]. Sentiment words are the words used to express the opinions, like *beautiful*, *awesome*, *bad*, and *awful*. There are sentiment phrases too, like *not bad*, *very interesting* and *costs an arm and a leg*. The task here is to identify the semantic orientation of a sentiment word or phrase, and to calculate its intensity; for example it seems that *awesome* is more powerful than *good*. Finding semantic orientations of words is instrumental in opinion mining [1]. It enables various methods to take advantage of it and analyze what people say in web 2.0 sites.

Since most sentiment words are adjectives with or without adverbs, The method introduced here is based on finding candidate sentiment phrases by searching for adjectives, and finding their term frequency in reviews on Amazon.com, and the overall score the reviewer has given to the product. This helps in processing reviews and opinions that are expressed in them.

The idea is that when a user gives a rating of 3 out of 5 stars to a product, he or she tends to use more moderate and less intense sentiment phrases, than when giving the whole 5 stars. So, the reviews are divided into five groups, each according to the number of stars the reviewer has given to the product.

The rest of paper is as follows: in section 2, the previous works that are done in this field are reviewed, in section 3 the algorithm is presented in detail, section 4 is for experiments, results and discussions, and section 5 concludes the paper with ideas for future works.

II. Previous Works

The works done in the field of world-level sentiment analysis can be grouped in two [2]: corpus-based methods, and dictionary-based approaches. Corpus-based approaches use large datasets, usually to find the relationship between opinion words; while dictionary-based methods use the lexical relations between words to find their orientations.

Hatzivassiloglou and McKeown [3] used conjoined adjectives of a large corpus to determine the orientation of these words. They used a clustering algorithm to distinguish different orientations.

Wilson et al. extended this work [4] to find more complicated relations between the words. In [5], the authors used a small set of seeds, and then calculated the score of other adjectives based on their distance from the previously known ones in the corpus.

In [6], a set of seeds is used and the score of other adjectives is calculated with Pointwise Mutual Information (PMI).

Baccianella et al. [7] proposed SentiWordNet 3.0, an enhancement of their previous work. They used the famous WordNet lexical database [8] and its relations for finding the orientation of sentiment words. They assigned three scores, Obj, Neg and Pos to each word; which correspond to the objectivity, negativity and positivity of the word, respectively and are numbers between 0 an 1.

Kamp et al. [9] also used WordNet. They calculated the distance of a word to "Good" and "Bad" to find if it is closer to the former or the latter. Xu et al. [10] proposed an algorithm named S-HAL, or Sentiment HAL, based on a previous algorithm called HAL which was presented by Lund and Burgess [11].

Qiu et al. [12] used the relationship between opinion words and topics or aspects to find new opinion words, and called it double propagation. Özsert and Özgür [13] linked WordNets of various languages, and used the relations to find the polarity of words.

In [14], the authors used an unsupervised framework to generate a subjectivity lexicon on the blogosphere.Maks and Vossen [15], a lexicon model is proposed to find the description of nouns, verbs and adjectives.

Na et al. [16] addressed the problem of domain-dependency in creating a lexicon. Rao and Ravichandran used label propagation to determine polarity of sentiment words [17]. Huang et al. [18] used constrained label propagation to automatically generate a domain-specific sentiment lexicon. Liang et al. [19] used a dependency expansion model to generate a sentiment lexicon.

III. The Aspire Algorithm

The proposed ASPIRE (Average Sentiment Polarity and Intensity Rating Extraction) algorithm tries to find values for some features for sentiment phrases, label the phrases as positive or negative, and do a classification task. The algorithm has two phases:

A. *Creating the Dataset*

The dataset is created by browsing a review site in which people give an overall rating to the subject and write a review about it, and choosing reviews randomly, with a uniform distribution, so that each rating group has the same number of reviews.

Here the reviews on cameras from Amazon.com are considered. Each review in this site has a rating of 1 to 5. So five groups must be created to assign each review to one of them.

Then the dataset is fed to a part-of-speech tagger; the one used here is The Stanford PoS-Tagger [20]. The main reason is that the opinion words are most likely expressed in the form of adjectives, or adverbs and adjectives. However, there are other forms of sentiment phrases without using adjectives, though they are not discussed here.

B. *Evaluating the Sentiment Phrases*

Here each sentiment phrase is considered to be an adjective, like *good*; or adverb plus adjective, like *very good*; or adverb plus adverb plus adjective, like *not very good*. This can capture more complicated sentiment phrases that cannot be evaluated by some other methods. There are other forms of sentiment phrases, but for simplicity only these three forms are considered.

Then score of a sentiment phrase is the first feature to introduce here. It is based on a weighted average, and is calculated using the formula (1):

$$score(n) = \frac{\sum_{s=1}^{5} freq(n,s) * s}{\sum_{s=1}^{5} freq(n,s)} \quad (1)$$

in which *s* represents each of the five groups and can take a value of 1, 2, 3, 4, or 5, based on the stars given. *n* is the word for which the score is calculated, and *freq(n, s)* is the frequency of the *n* in the group *s*. For example, *freq(good, 4)* shows the frequency of the word *good* in the 4 starred reviews. This weighted average is a number between 1 and 5. The intuition is that the closer the score to 5, the more positive the word, and the closer the score to 1, the more negative the word. Here we want to check whether this is true or not.

Then, the extracted sentiment phrases are divided into two classes: *Positive* and *Negative*. Each sentiment phrase is labeled by hand. Then a set of features are created using *score(n)* and *freq(n, s)* so that the problem is turned into a classification problem.

The other features here are as follows. The second feature to be used is named as Normalized Frequency, and is shown in the formula (2):

$$NF(n,i) = \frac{freq(n,i)}{\sum_{s=1}^{5} freq(n,s)} \quad (2)$$

This formula which shows the normalized frequency of a sentiment phrase in a group, so that the dependency on the size of dataset is dropped and it does not matter whether a sentiment phrase is a popular one and is used extensively, or just a few times. It is clear that NF lies between 0 and 1, and the sum of NFs for a noun in each group is equal to 1. For each i (which can take 1, 2, 3, 4, and 5 as values), NF(n, i) is considered as a feature for sentiment phrases.

Another feature to be introduced is named *Divergence* and shows the uniformity of *NF* in different groups. If the second

maximum is shown as m*ax2* and third maximum is shown as m*ax3*, the formula for Div_2 and Div_3 features are as follows:

$$Div_2(n) = \max_i NF(n,i) - \max 2_i NF(n,i) \quad (3)$$

$$Div_3(n) = \max_i NF(n,i) - \max 3_i NF(n,i) \quad (4)$$

Div_2 shows if for a phrase, the NF is high for one group and low for other groups. For example, it seems that phrases like "very great" will be found much more in 5-starred reviews than other groups and Div_2*(very great)* will have a value close to 1. But Div_3 is a more accurate measure. If NF(n, i) is high, it is likely that NF(n, i-1) or NF(n, i+1) is high too. A sentiment phrase like "excellent" is mostly found in 5-starred groups but its frequency in 4-starred groups is not low, though it is rarely seen in other groups. Hence Div_3 is arguably a better measure.

Then, using a C4.5 decision tree and these features, the classification problem can be solved.

IV. Experiments and Evaluation

The dataset was created of various camera reviews on Amazon.com. Of each group of ratings, 70 reviews were collected randomly. So, a total of 350 reviews made their way into the dataset. Some reviews were as short as 30 words, and some other used more than 800 words. Overall, the length of dataset was about 110,000 words.

After the implementation of ASPIRE, the results needed some post-processing; they needed some pruning to be done, because of the obvious fact that not every single adjective is a sentiment phrase. For example, the word *initial* is an adjective, but it does not have a definite polarity. Another potential problem is that words like *good* are used to express opinions about objects and their aspects, but they can be used in other sentences too, like: *"I have a good idea: do not buy this camera!"* But since all the adjectives, more or less, can be used this way, the effect can be ignored.

The C4.5 decision tree showed that the feature *score(n)* is all that is needed for classification of sentiment phrases, and the other features, like *Div2* and *NF* cannot do better than *score*. The threshold was 2.5, so sentiment phrases with a score of 2.5 or lower were considered negative, and those with scores higher than 2.5 were positive ones.

However, by eliminating *score* it was observed that the feature *NF(n, 1)* has the second highest impact on deciding whether a sentiment phrase is negative, or positive.

The method for validation was 10-fold cross-validation which splits the dataset to 10 subgroups, runs for 10 times and each time, one of the subgroups is considered as the test set and the remaining 9 subgroups form the training set. The confusion matrix of can be seen in Table 1.

TABLE I. CONFUSION MATRIX FOR THE CLASSIFICATION

		Actual Class	
		Positive	**Negative**
Predicted Class	**Positive**	144	34
	Negative	4	93

So, precision, recall and f-measure for the classes, and overall accuracy are as follows:

Precision(Positive) = 80.90%

Precision(Negative) = 95.88%

Recall(Positive) = 97.30%

Recall(Negative) = 73.23%

F-Measure(Positive) = 88.34%

F-Measure(Negative) = 83.04

Accuracy = 86.15% ± 4.32%

Table 2 shows some top rated sentiment phrases, and their score. It can be concluded from the results that words with higher rankings seem to be more positive.

TABLE II. SOME OF THE HIGHEST RATED SENTIMENT WORDS

Sentiment Word	***Score***
Superb	4.83
Magical	4.8
So Easy	4.75
Comfortable	4.67
Favorite	4.63
Pleased	4.54
Affordable	4.5
Really Great	4.5
Fantastic	4.43
Pretty Good	4.22

And some of the lowest rated sentiment phrases are shown in Table 3.

TABLE III. SOME OF THE LOWEST RATED SENTIMENT WORDS

Sentiment Word	*Score*
Notorious	1.00
Irritating	1.00
Very Disappointed	1.67
Extremely Slow	1.67
Very Very Slow	2.00
Completely Terrible	2.00
Defective	2.09
Frustrated	2.25
Loud	2.36
Unacceptable	2.42

Some scores for both positive and negative classes are compared to SentiWordNet in Tables 6 and 7. SentiWordNet shows the score of each word in an ordered triad, in which the first, second and third component show positivity, objectivity and negativity, respectively. The best and closest result in SentiWordNet is considered here.

TABLE IV. COMPARISON OF POSITIVE SCORES

Words / Score	*ASPIRE*	*SentiWordNet*
Creative	4.15	(0.375, 0.625, 0)
Fast	4.21	(0.25, 0.75, 0)
Fantastic	4.43	(0.75, 0.25, 0)
Special	3.27	(0.25, 0.75, 0)
Affordable	4.5	(0, 1, 0)
Small	3.86	(0, 0.875, 0.125)
Incredible	3.89	(0, 1, 0)
Acceptable	4.17	(0.625, 0.375, 0)
Versatile	4.37	(0.25, 0.5, 0.25)
Cool	4.17	(0.375, 0.625, 0)

Words like *fast* are very positive in describing a camera. The word *fast* has a score of 4.21 in ASPIRE. But SentiWordNet does not show this intensity, because of its generality and domain-independency, and cannot show its significance in the camera domain. The word *affordable* is a compliment (with a score of 4.5 here), though its positivity cannot be captured by the means of WordNet and it is considered as a completely objective word. The same can be said about *incredible*. The word *cool* is also rather informal, and is much more positive than its ratings on SentiWordNet suggests.

TABLE V. COMPARISON OF NEGATIVE SCORES

Score of Words	*ASPIRE*	*SentiWordNet*
Awful	2.00	(0, 0.125, 0.875)
Terrible	2.36	(0, 0.125, 0.875)
Faulty	1.5	(0.5, 0.5, 0)
Unreliable	1.5	(0, 0.125, 0.875)
Unacceptable	2.42	(0, 0.25, 0.75)
Disappointing	2	(0, 0.25, 0.75)
Abysmal	2	(0.25, 0.5, 0.25)
Klunky	2	N/A
Mangled	2	(0, 0.25, 0.75)
Defective	2.09	(0, 0.25, 0.75)

Here, faulty as "having a defect" does not have any negativity in SentiWordNet 3.0, though it is shown to have a very high negativity in ASPIRE. Some words, like *klunky* are too informal to appear on SentiWordNet 3.0., though they do show some opinion and cannot be ignored. The word *abysmal* in a more informal way is much more negative than what is suggested in the dictionaries.

Also, using ASPIRE can be helpful to understand better the role of some adverbs like "very" and "not".

Table 6 shows the effect of the word "very".

TABLE VI. THE EFFECT OF USING "VERY" IN PHRASES

Score of Words	*ASPIRE*
Accurate	4.05
Very Accurate	4.25
Easy	4.17
Very Easy	4.36
Versatile	4.38
Very Versatile	5.00
Loud	2.36
Very Loud	1.67
Good	3.68
Very Good	3.79

This table shows the intensifying effect of using "very". In some cases this effect is not that much, though in some words, like loud, the effect of using "very" is indeed very high!

The advantages of ASPIRE are described as follows:

a) ASPIRE can be domain-specialized. Having a domain-independent sentiment lexicon has its benefits, but has serious drawbacks. Some words are positive in one domain and negative or objective in other domains. The word fast is a

complement for a camera or a car, but is not a good opinion for a movie. Having specialized sentiment lexicons can lead us to have a better understanding on opinions, and process them more precisely than going with a universal, all-round lexicon.

b) ASPIRE can assess complicated phrases. Sentiment phrases like "Not Very Good" and "Very Disappointing" can be classified in ASPIRE as positive or negative. The effect of using *not* or *very* is much more than simply negating or intensifying the adjectives. If bad has a score, not bad does not necessarily have a complement score. And phrases like *not very good* are even harder to interpret. Also using an improved method, implicit sentiment phrases like "Costs an arm and a leg" can be captured.

c) ASPIRE can make us have a better understanding of the informal language. Some methods are based on dictionaries and formal definitions, but the actual language that people use is different. In SentiWordNet, the word "special" is purely objective or a little positive, but using this word in a sentence shows the pleasure of the reviewer: "This camera is special!" Some words cannot even be found in dictionaries. The word *klunky* is an example.

d) ASPIRE can show the changes in trends through time. The meanings of words constantly change, and some words are trendy. They may have a great impact today, but tomorrow they can be forgotten or lose their intensity. With methods like ASPIRE that assess the words more precisely, these changes can be followed.

e) ASPIRE can show the polarity of some phrases that are assumed to be objective and do not contain any sentiments. Words like *sufficient*, *dynamic*, and *practical* are more or less considered objective. Some more objective phrases, like *aware*, *live*, *subtle* and *digital* have high scores too. This is in part because when people use these words they want to describe something good rather than bad.

v. Future Works

The results of ASPIRE are promising; but some adjustments can lead to better results.

The weighted average formula can be enhanced to give a more precise score. Also, it was observed that when an adjective is not a sentiment word, its distribution amongst the five groups of ratings is more uniform. So this can be a factor in filtering out the adjective that are not sentiment words.

Only three forms of sentiment phrases, based on the PoS-Tags of the words in sentences are considered. There are more forms and some of them require the sentence to be parsed.

The corpus was limited to the reviews of cameras. So, some domain-specific phrases are seen here, like "blurry". Sentiment words from different domains can be gathered to focus on more general and cross-domain sentiment words.

Use of adjectives in 5-star reviews are considerably more than in 1-star reviews. So, it seems that the adjectives used in the latter group should have more weight in calculating the overall score of each sentiment phrase. But how to give more importance to them? This is something to be studied later.

References

[1] B. Liu, "Sentiment analysis and opinion mining", Morgan & Claypool Publishers, 2012.

[2] M. M. S. Missen, M. Boughanem, and G. Cabanac, "Opinion mining: reviewed from word to document level." Social Network Analysis and Mining, 2013, pp. 1-19.

[3] V. Hatzivassiloglou, and K. R. McKeown, "Predicting the semantic orientation of adjectives", Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 1997, pp. 174–181.

[4] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis", In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2005, pp. 347–354.

[5] M. Baroni, and S. Vegnaduzzo, "Identifying subjective adjectives through web-based mutual information", In Proceedings of KONVENS, vol. 4, 2004, pp. 17–24.

[6] P. D. Turney, and M. L. Littman, "Measuring praise and criticism: Inference of semantic orientation from association", ACM Transactions on Information Systems (TOIS), vol 21.4, 2003, pp. 315–346.

[7] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining", LREC, Vol. 10, 2010, pp. 2200–2204.

[8] C. Fellbaum, WordNet, Springer Netherlands, 2010, pp. 231–243.

[9] J. Kamps, M. J. Marx, R. J. Mokken, and M. De Rijke, "Using wordnet to measure semantic orientations of adjectives", 2004, pp. 1115– 1118.

[10] T. Xu, Q. Peng, and Y. Cheng, "Identifying the semantic orientation of terms using S-HAL for sentiment analysis", Knowledge-Based Systems, Vol. 35, 2012, pp. 279–289.

[11] C. Burgess, and K. Lund, "Modeling cerebral asymmetries in high-dimensional semantic space", Right hemisphere language comprehension, 1998, pp. 215–244.

[12] G. Qiu, B. Liu, J. Bu, and C. Chen, "Expanding Domain Sentiment Lexicon through Double Propagation", In IJCAI, Vol. 9, 2009, pp. 1199–1204.

[13] C. M. Özsert, and A. Özgür, "Word polarity detection using a multilingual approach." Computational Linguistics and Intelligent Text Processing", Springer Berlin Heidelberg, 2013, pp. 75–82.

[14] H. W. Chen, K. R. Lee, H. H. Huang, and Y. H. Kuo, "Unsupervised subjectivity-lexicon generation based on vector space model for multi-dimensional opinion analysis in blogosphere", Advanced Intelligent Computing Theories and Applications, Springer Berlin Heidelberg, 2010, pp. 372–379.

[15] I. Maks, and P. Vossen, "A lexicon model for deep sentiment analysis and opinion mining applications", Decision Support Systems, Vol. 53.4. 2012, pp. 680–688.

[16] S. H. Na, Y. Lee, S. H. Nam, and J. H. Lee, "Improving opinion retrieval based on query-specific sentiment lexicon", Advances in Information Retrieval, Springer Berlin Heidelberg, 2009, pp 734–738.

[17] D. Rao, and D. Ravichandran, "Semi-supervised polarity lexicon induction." In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, 2009, pp. 675-682.

[18] S, Huang, Z. Niu, and C. Shi, "Automatic construction of domain-specific sentiment lexicon based on constrained label propagation." *Knowledge-Based Systems* 56, 2014, pp. 191-200.

[19] J. Liang, J. Tan, X. Zhou., P. Liu, L. Guo, and S. Bai, "Dependency Expansion Model for Sentiment Lexicon Extraction." Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on. Vol. 3. IEEE, 2013.

[20] http://nlp.stanford.edu/software/tagger.shtml