Machine Reading for Notion-Based Sentiment Mining

Roula Hobeica[1] Hazem Hajj[1] Wassim El Hajj[2]

[1]Department of Electrical and Computer Engineering

[2]Department of Computer Science

AUB

Beirut, Lebanon

rah70, hh63, we07@aub.edu.lb

***Abstract*—Although several sentiment classification methods have been proposed, rare are the ones that provide a solid link between human analysis of a sentiment text and machine analysis of the same text. In this paper, we investigate the automation of human's reading and building of notions. We show that the proposed process of automated machine reading and notion extraction can be used for sentiment mining and the identification of product characteristics. We then focus on the issue of sentiment classification of online reviews which have been receiving an increased level of attention by online users as well as data mining researchers in the past few years. Our proposed method bases itself on the human's thought process in extracting notion and sentiment from text. The approach first employs part of speech (POS) tagging, then learns the product features and characteristics that form the preconceived thoughts or notions about a topic's sub features. The idea is similar to how humans learn and build preconceived notions based on the review's topic, and then use it when reading a new review. Experiments show the success of the method in sentiment mining and in extracting a product's desirable features.**

Keywords: Machine reading, sentiment mining, notions, feature extraction

I. INTRODUCTION

At the onset of the internet age, huge amounts of data were being made available to users online. At that time, the data found was mostly objective. As the internet grew even more, social networking sites, blogs, forums, and tweets, [1] gave users the ability to express themselves and their sentiments in even more ways, and regarding any topic they have in mind.

Having billions of texts representing people's sentiments scattered around the Web, and thousands of business applications based on those opinions, it is clear that the task of analysis and sentiment extraction from this huge number of texts cannot be done manually. For this reason, it is vital to have a framework optimized for the efficient gathering and processing of relevant data and sentiments from text. The goal of such a system would be to produce results comparable to those a human would produce, however, in much less time. Although there have been major improvements in the field of sentiment mining and analysis, many challenges remain including mining sentiments in languages such as Arabic, feature selection [2], and result assessment [3]. Moreover, one of the main reasons that causes machine-based sentiment mining techniques to be fragile is the fact that no clear ideal approach has been suggested specifically for sentiment analysis of text, whether one is dealing with sentence-level or document-level sentiment analysis.

In this paper, we propose a method to build background knowledge from the available online reviews, and then use the knowledge to assess sentiment in new reviews. To further emphasize the importance of background information in understanding a passage's implied meaning, we refer to studies made in the field of education, and specifically in the area of literacy education. In [4], the author mentions: "The ability of fluent readers to integrate text and background information appropriately and efficiently is the hallmark of expert reading in a topical domain (e.g. history, biology, psychology)." In particular, we examine how preconceived notions can be automatically learned to build the needed background knowledge. For example, if one is reading a review about cars, and the phrase "it needs constant maintenance" appears, that person would link the phrase to the notion that having to maintain a car constantly is a bad thing and so would know that this part of the review is negative. In contrast, if that person had no previous background about the review's topic, he would not know whether this statement is positive or negative, especially since sentiment words do not exist in this phrase.

Our proposed method is inspired by observing how humans use their preconceived notions when reading. As humans have tailored their reading abilities for quick understanding and interpretation of text, and especially for the detection of sentiments implied within documents, it is important to know the underlying reading and comprehension mechanisms that take place when a human is reading a new passage. According to [4], and as shown in Fig. 1, there are two main components that are involved during the reading process.

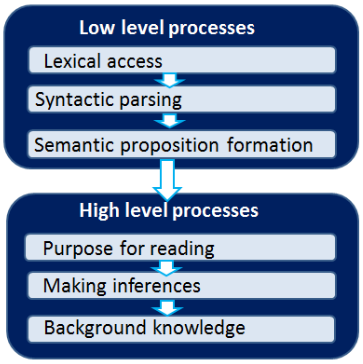

Figure 1. Human reading process

The first component encompasses low-level processes such as lexical access, which is direct recognition of a word's meaning, syntactic parsing, which is the process by which the reader parses the passage grammatically, as well as semantic proposition formation which bases itself on syntactic parsing and lexical access to form sub-sentence structures called clauses, with each clause representing a certain idea or concept. The second component composed of high level processes plays the major role in reading. It includes having a defined purpose for reading, making inferences, as well as extensive use of background knowledge.

In this respect, when people read a review, they automatically infer the topic of that review, either since they queried about that specific topic, or by deducing the topic from keywords used in the review. Having deduced the topic while reading, the person starts to link each phrase being read with some preconceived notions about the topic or subtopics being discussed, where a notion is mainly a topic's characteristic along with its associated sentiment according to that person. These notions would enable the person to decide whether the review is positive or negative.

The above-mentioned human reading steps provide the foundation flowchart (see Fig. 1) to achieve automated machine reading, notion extraction, and ultimately sentiment mining. The rest of the paper is divided as follows: Section 2 presents related work in the field of text sentiment mining. Section 3 introduces and presents the proposed approach. Section 4 describes the experiments and corresponding results. Section 5 concludes the paper and highlights future work.

II. Previous Work

Several approaches have been suggested for sentiment mining. Some targeted the optimization of feature selection [2], while others tried to compare results by using different learning and/or non-learning algorithms. Some researchers also dealt with sentiment analysis in languages such as Arabic [5] and Chinese [6]. In addition to that, several authors came up with new applications for sentiment mining such as prediction of sales performance using the classified online sentiments.

In [7], Bing Lui proposes to define an object by a quintuple that includes the object itself, a feature of that object, the orientation of the opinion on that feature, the opinion holder, as well as the time during which the opinion was made. Liu also mentions some of the biggest problems in the sentiment mining field, such as named entity recognition, coreference resolution, and feature extraction. With respect to the latter, many feature selection approaches have been proposed in the literature. In [8], the authors represent movie reviews by feature vectors composed of corresponding appraisal word [9] frequencies, whereas the authors of [10] used TFIDF for feature representation of online stock message boards. In [11], the feature dimension used consisted of sentiment scores of words. Those scores were calculated based on each word's part of speech (POS) tag, using the Stanford POS tagger [12] as well as an average score computed by making use of SentiWordNet [13]. A simple way of representing a video game review was implemented in [14] and [15], where the authors used Boolean feature vectors representing the status of a word in each review (i.e. present or absent). Authors of [14] used words that show the most correlation to review scores as features. On another level, and in order to deduce the sentiment orientation of a document, the authors of [5] use the sentiment orientation of a document's sentences as a feature vector. Seeking to find the optimal set of features in a sentiment analysis problem in order to avoid over-fitting, the author of [2] proposes an Intelligent Feature Selection Method (IFS) which uses semantic and syntactic information to refine large input feature spaces by removing redundant features according to their degree of correlation.

Many are the methods that were used for sentiment mining. While most were based on machine-learning schemas, such as SVM [5], [11], [14-15], Naïve Bayes [10], [14-15], and Decision Trees [10], some authors made use of different techniques such as Probabilistic Latent Semantic Analysis (PLSA) [8], Latent Dirichlet Allocation (LDA) [16], and Pointwise Mutual Information-Information Retrieval (PMI-IR) [17].

Throughout the mentioned literature, the sentiment classification results were mostly reported in terms of accuracy, and ranged between 82.9% [15] and 90.7 % [2]. Other measures used were mean average percentage error (MAPE) [8], F-measure which is a combination of recall and precision [6], as well as comparisons against defined baseline methods [16] and average mean square error (MSE) [14].

From all the above mentioned work, one could extract useful tips for obtaining a solid sentiment analysis system. Accordingly, the latter should follow the policy that having more features is not always better [4]. Moreover, domain-tailored features could increase the sentiment analysis accuracy in some applications [10]. However, the mentioned approaches have many limitations including the lack of a strong argument basis that would lead to an optimal choice of features such that the sentiment analysis accuracy could increase over multiple (or all) domains. Consequently, the following proposed scheme will try to address that weakness.

III. Proposed Scheme

Although machines might outperform humans in certain complex classification tasks, it is still obvious that they are very far from extracting sentiments from text the same way a human would. Our proposed method thus stems itself from the thought process a human would unconsciously follow in order to read and deduce a review's implied sentiment. Consequently, this process could be seen as the automated algorithm's role model, implying that the closer an algorithm replicates the human thought process, the more accurate will its sentiment classification results be. Accordingly, the workflow presented in Fig. 2 is followed by our proposed machine reading process in an attempt to replicate the basic human reading and analysis process.

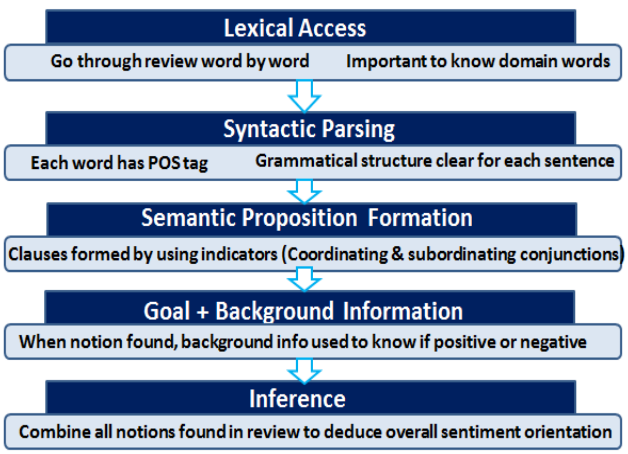

Figure 2. Machine reading process

Based on the machine reading process, our method is able to achieve three objectives; the first being feature extraction, where features are defined as the nouns and verbs that make up a domain's main lexicon. The second objective is the extraction of notions from a domain's reviews, which provides background knowledge of desirable product qualities versus undesirable ones. The third objective is the usage of background notion information to infer the overall sentiment of a given review. All three objectives are presented below.

A. Feature Extraction

Extracting the domain's features is, in its essence, analogous to building the domain-specific lexicon which will be used during the machine reading process at the lexical access level. The algorithm describing the feature extraction details is shown in Fig. 3, and is driven by the principle that when many reviewers mention the same feature in their reviews, one can imply that the feature in question affects a person's view on the product.

```
Inputs: x: threshold, tr: tagged review, db: database

While (still have untrained tr)
    Read tr
    Divide tr into words
    Foreach word
        If (POS == Noun or POS == Verb)
            Store in database
        Else
            Increment word frequency in db
    End foreach
End while
Choose nouns and verbs with frequency > x and store as domain features
```

Figure 3. Feature extraction algorithm

At this level, the algorithm would have acquired background information on the domain in terms of what product features are important to people (through the domain nouns) as well as what product behavior is important to people (through the domain verbs). More details on those results will be shown in a later section.

B. Notion Extraction

Since notions have no specific structure, in the sense that no clear rules define them grammatically, heuristic-based rules were used to extract notions from the reviews. Consequently, it was defined that in each single clause, only one notion can be found. Moreover, a notion was defined to be formed of sequential combinations of nouns, verbs, adverbs and adjectives given that it has to be composed of more than one word. Fig. 4 presents our notion extraction algorithm.

```
Inputs: tr: tagged review, db: database

While (still have untrained tr)
    Divide tr into clauses
    Foreach clause
        Extract main idea from clause
        If (main idea contains domain features)
            If (idea not found in notions db)
                store idea as notion in db
                notion score = review rating
            End if
            Else
                update notion score in db = S = (Σ_{i=1}^{i=n} r(i)) / n
        End if
        Else
            main idea is not a domain notion => disregard it
    End foreach
End while
```

Figure 4. Notion extraction algorithm

As depicted by the above algorithm, and to mimic a reader's syntactic parsing process, our proposed algorithm parses the tagged reviews, and identifies the parts of speech of each text's encompassed words. Having done that, and to perform semantic proposition formation, sentences are divided into clauses according to the presence of known coordinate conjunction and subordinate conjunction words such as and, but, for, or, so, after, although, because, and others [18].

However, not all clause ideas are considered as notions in the domain being tackled; in the sense that some clauses could present a very general idea that would be irrelevant in building a domain's background notions. This is where using the domain's lexicon previously built comes in very handy, keeping in mind the fact that the high frequency of a certain noun or verb across many reviews will indicate that it is closely related to the domain and thus, a notion that includes that domain feature would have an impact on the overall sentiment of the review. Consequently, notions are formed by going through the training reviews, and only storing notions that contain at least one domain feature, be it noun or verb.

As each notion is formed and stored in the notion database (using the known review ratings from the training dataset), it is assigned a score on a scale of 1 to 5 indicating its degree of positivity or negativity according to the following equation:

$$S=\frac{\sum_{i=1}^{i=n} r(i)}{n} \qquad (1)$$

Where S is the notion sentiment score, n is the number of reviews that contain the notion, and r(i) is the rating of review i which is an integer from 1 to 5 assigned by reviewers in order to represent their view of the product according to the scoring scale.

C. Sentiment Mining

In order to classify a review being tested as positively or negatively oriented, notions are extracted from the review the same way they were extracted and stored in the notion database during the training phase. If those notions are found in the database, the review is assigned a score which is the average of the scores of its contained notions. The overall sentiment mining process is presented in Fig. 5.

```
Inputs: tr: tagged review

While (still have unclassified tr)
    While (not end of tr)
        Extract notions from tr
        If (notion is a valid domain notion)
            Update review score by adding notions score
        End if
    End while
    final review score = review score / number of notions in review
End while
```

Figure 5. Sentiment mining algorithm

The obtained score is then rounded to the nearest integer. Consequently, reviews with a score of 1, 2, or 3 are classified as negative reviews, whereas those with a score of 4 or 5 are classified as positive reviews.

IV. IMPLEMENTATION

Due to the importance of POS tagging for notion-based sentiment analysis, the first step of the proposed method was to use the "bidirectional-distsim" Stanford POS tagger [12] that achieves an accuracy of around 97.1% for the English language. Consequently, 5000 reviews on digital portable audio devices were extracted from Epinions [19] by using RapidMiners's Web Crawl Operator [20], and then tagged. Those reviews were rated from 1 to 5 stars, 5 being the best rating, and contained 488 1-star reviews, 386 2-star reviews, 1728 4-star reviews, and 2398 5-star reviews. Reviews with a 3-star rating were omitted since they represent neutral opinions that would compromise our notion scoring process. Having done that initial step, we start our training procedure as detailed in Fig. 6. The programming language of choice was PHP which provides text-handling capabilities, and an easy interaction with MySQL database to efficiently store any needed data.

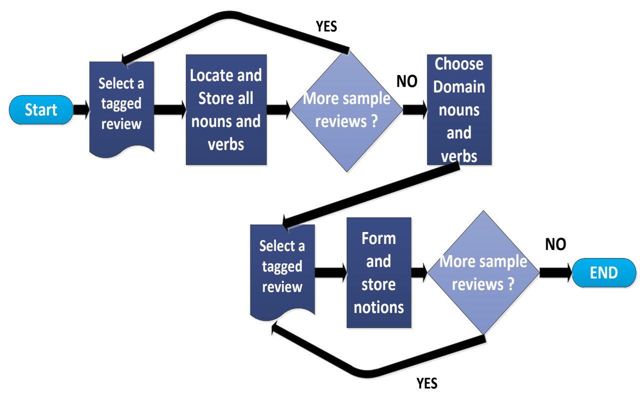

Figure 6. Main Flow of Processes

A. Feature Extraction Results

Based on the approach suggested in section 3, each review was parsed, and its contained nouns and verbs were stored in a table to represent the domain features, where the domain in this case was digital cameras. This feature extraction step produced 7007 nouns, however, only those with a frequency above 119 were kept, resulting in 277 feature nouns. Whereas for verbs, the ones with a frequency greater than 5 were kept, resulting in 1585 feature verbs. The frequency thresholds chosen for this purpose were selected heuristically according to the observed results. Table I shows the 10 most frequent nouns that were found in one of the training cycles. It is very clear that the collected features demonstrate part of the digital cameras' domain-specific lexicon.

TABLE I. MOST FREQUENT NOUNS

Noun	Frequency
camera	24122
picture	9951
battery	4591
quality	3535
photo	3289
time	2780
feature	2756
mode	2744
shot	2576
card	2567

Other frequent nouns also included flash, zoom, memory, screen, as well as many other digital camera features that interest consumers in general and would thus lead reviewers to base their opinion on the camera according to how poor or great the camera, in general, and its features, in particular, performed. Here it is worth noting that all nouns were stemmed by using the PlingStemmer which is part of the javatools [21] classes, and was especially designed to stem English nouns. By stemming the nouns, we joined nouns that are essentially the same together. For example, when found in a review, the noun "batteries" was stemmed and resulted in the noun "battery", thus correctly increasing the frequency of the feature "battery" in the table.

B. Notion Extraction Results

Following the feature extraction step, we extract the Notions from the training reviews as was described in section 3.2, and as a result, around 12,000 notions were obtained. However, those notions were pruned by removing notions that have a frequency of 1 as well as a score of 5, since many of them might not reflect true positivity, especially since the number of 5-star reviews in the training set is very high. Table II shows the results of some frequent positive and negative notions.

TABLE II. FREQUENT NOTIONS

Notion	Score
Look elsewhere	1.0
Not recommend camera	1.0
Warranty repair	1.0
Stopped working	1.0
Shutter speed	5.0
Love camera	5.0
Is great camera	5.0
Wide angle	5.0
Is very compact	4.22
Color are bright	4.66

According to the above results, it can be clearly seen that desired features in a camera are: having a wide angle and bright colors, whereas a camera that stops working and is not recommended are repelling features.

C. Sentiment Mining Results

When all the notions had been extracted, testing was performed. The testing procedure is based on comparing the notions formed during the training phase with the notions found within the review being tested. The sentiment score of the latter would thus increase if it includes one of the stored positive notions, and decrease if a stored negative notion is found within it.

The evaluation of the proposed method's results was performed using the 5-fold cross-validation method. By using this validation scheme, the testing would be more accurate since the samples used for training and testing are interchanged and chosen randomly. To implement this method, each group of equally rated reviews was split into 5 equal parts and each part was assigned to one of the 5 groups. When one group (~ 1000 elements) was used for testing, the other groups (~ 4000 elements) were used for training. An average accuracy of 85 % was recorded for our notion-based approach whereas a highest accuracy of 87.01 % was achieved in one of the testing rounds.

Accordingly, Table III presents the Confusion Matrix [18] for the testing round that achieved the best results. Hence, the Confusion Matrix [22] represents the distribution of the classification results in terms of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN).

TABLE III. CONFUSION MATRIX

Result \ Truth	Positive	Negative
Positive	802	106
Negative	24	69

From the confusion matrix above, the accuracy of the notion-based sentiment classification system can be calculated. Since accuracy has been used as an evaluation metric in several previous publications [2] [10] [11] [17] [15] [5], it is beneficial to calculate it for our notion-based sentiment classification approach. Hence, the formula below was used to calculate accuracy.

$$\text{Accuracy}=\frac{(TP+TN)}{(TP+TN+FP+FN)}=\frac{(802+69)}{(802+69+106+24)}=87.01\ \% \qquad (2)$$

Achieving the above accuracy indicates that our proposed machine reading technique is a promising approach that is worth more thorough investigation. We believe that the suggested approach can be highly accurate and efficient in inferring sentiments from text when internally optimized or combined with other sentiment extraction methods presented in the literature.

V. CONCLUSION

Although important achievements in the field of sentiment mining and analysis have been reported in the literature, no approach has based itself on the thought and decision process of a human when analyzing sentiments. Since the optimal sentiment analysis is the one that occurs in the human brain, we presented in this paper a novel notion-based sentiment classification system that tries to mimic a human's acquired notions about a certain topic and their influence on his understanding of written sentiments whether they are explicitly stated or not. When tested on a set of 5000 reviews, our notion-based approach produced accurate results; the fact that proves the merit of our suggested approach. We do believe though that a lot can be done to increase the current accuracy even more. One direction will be to optimize the formation of notions by studying the English language sentence-level grammar in more depth. Notion detection will be complemented by synonym detection so that words indicating the same notion can be combined together. Moreover, storing stemmed notions will be studied to check if it contributes positively to the overall accuracy. Another direction will be to optimize the threshold values used in our approach, for instance, the thresholds used in the feature selection stage. A third direction will be to investigate the benefit of combining our approach with current state-of-the art methods. In summary, a lot of improvements can be made to enhance the already acceptable accuracy of the suggested model.

ACKNOWLEDGMENT

This research is supported by the Lebanese National Council for Scientific Research (LNCSR), the American University of Beirut, and Intel MER fund.

REFERENCES

[1] Hsinchun C.; Zimbra, D.; "AI and Opinion Mining," Intelligent Systems, IEEE, vol.25, no.3, pp.74-80, May-June 2010

[2] Abbasi A., "Intelligent Feature Selection for Opinion Classification" Intelligent Systems, IEEE, vol.25, no.4, pp.75-79, July-Aug. 2010.

[3] Esuli A.; Sebastiani F., "Sentiment Quantification" Intelligent Systems, IEEE, vol.25, no.4, pp.72-75, July-Aug. 2010

[4] W. Grabe and F. L. Stoller, Teaching and Researching Reading. London: Pearson Education Longman, 2011

[5] Farra, N.; Challita, E.; Assi, R.A.; Hajj, H.; "Sentence-Level and Document-Level Sentiment Mining for Arabic Texts," Data Mining Workshops (ICDMW), 2010 IEEE International Conference on , vol., no., pp.1114-1119, Dec. 2010

[6] Qingliang M., Qiudan L., Ruwei D., "AMAZING: A sentiment mining and retrieval system" Expert Systems with Applications, vol. 36, Issue 3, Part 2, pp. 7192-7198, 2009

[7] Bing L.; "Sentiment Analysis:A Multifaceted Problem" Intelligent Systems, IEEE, vol.25, no.3, pp.76-80, May-June 2010

[8] J.E. Bourne. "Blog Data Mining: The Predictive Power of Sentiments," in Data Mining For Business Applications, Longbing Cao, Philip S. Yu, Chengqi Zhang, Huaifeng Zhang. New York: Springer, 2009, pp.184.

[9] Whitelaw C., Garg N.,; Argamon S. Using appraisal groups for sentiment analysis. In CIKM ' 05, pp. 625–631, 2005

[10] Sehgal, V.; Song, C.; "SOPS: Stock Prediction Using Web Sentiment," Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on , vol., no., pp.21-26, 28-31 Oct. 2007

[11] Yan D.; Yulei Z.; Hsinchun C.;"A Lexicon-Enhanced Method for Sentiment Classification: An Experiment on Online Product Reviews," Intelligent Systems, IEEE , vol.25, no.4, pp.46-53, July-Aug. 2010

[12] Toutanova K., Klein D., Manning C., Singer Y., StanfordPOStagger,[Online]Available:http://nlp.stanford.edu/software/tagger.shtml,Stanford, 2003

[13] Esuli A., Sebastiani F.," SENTIWORDNET: A publicly available lexical resource for opinion mining " Language Resources and Evaluation (LREC), 2010 Conference on, p. 417—422, 2010

[14] Drake, A.; Ringger, E.; Ventura, D.; "Sentiment Regression: Using Real-Valued Scores to Summarize Overall Document Sentiment," Semantic Computing, 2008 IEEE International Conference on , vol., no., pp.152-157, 4-7 Aug. 2008

[15] Pang B., Lee L., Vaithyanathan S.; "Thumbs Up? Sentiment Classification Using Machine learning Techniques'" Empirical methods in natural language processing EMNLP '02 ACL-02 conference on, vol 10, pp. 79-86

[16] Celikyilmaz, A.; Hakkani-Tür, D.; Junlan Feng; , "Probabilistic model-based sentiment analysis of twitter messages," Spoken Language Technology Workshop (SLT), 2010 IEEE , vol., no., pp.79-84, 12-15 Dec. 2010

[17] Turney P.D; "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews" Annual Meeting on Association for Computational Linguistics, (ACL), 2002, pp 417--424. Morristown, 2002

[18] R.P. Craig and V. F. Hopper, Barron's 1001 Pitfalls in English Grammar. New York: Barron's Educational Series, Inc., 1986

[19] Available: http://www.epinions.com/Digital_Cameras

[20] Rapidminer [online] Available: http://rapid-i.com/

[21] Javatools [online] Available: http://mpii.de/yago-naga/javatools

[22] Kohavi, R. "Glossary of Terms" (R. Kohavi & F. Provost, Eds.)Machine Learning, December 1998, pp. 271-274