Cerebro: A Software Service for Identifying Digital Marketing Opportunities

[1]Daniel J. Y. Lee and [2]Kok-Leong Ong

Abstract

We present Cerebro as a software service (SaaS) utilizing a number of novel concepts for businesses to monitor voices in the online social sphere via social sensors. Using information collated from these social sensors, Cerebro constantly identifies social conversations, topics and news that are deemed positive and conducive for digital marketing opportunities. Unlike keyword-based advertisement models, which are insensitive to the search query or the content's sentiment, Cerebro provides near real-time identification of digital marketing opportunities from evolving pockets of positive sentiment to a product or service. This approach enables a more effective marketing campaign than keyword-based advertisements over time as it allows businesses to vary its campaign destination and target groups quickly to reflect the evolving sentiment in the digital environment. Cerebro's SaaS architecture is designed to be scalable while being easily accessible as a Web Service. A current running prototype is now available for public evaluation.

1. Introduction

In recent years, a new facet of the Web is emerging and has reached critical mass. Driven primarily by sites such as FaceBook, YouTube, Twitter, etc., they all share a common characteristic. That is, they carry a large volume of user-generated content and are now responsible for a significant proportion of traffic on the Internet.As these new avenues overtake traditional media such as newspaper and television on its viewership, a flow on con- sequence can be seen in many areas.In the case of advertising, businesses are now increasingly adjusting their budgets to account for online marketing campaigns.

*CorrespondingAuthor:
(E-mail:).
[1]School of Information and Business Analytics, Deakin University
70 Elgar Road, Burwood, Victoria 3125, Australia

While generally it has been the case that budgets for online marketing campaigns are growing, nevertheless they remain limited and finite. Such as, how to maximize the finite resources to achieve the best outcomes remain an important question. According to [2] keyword-based marketing (e.g., Google Adwords) is the most popular form of online marketing tool, search engine optimisation, keyword discovery and selection become tools maximize outcomes from the budget. Until now, this approach has been generally successful. Nevertheless, the model has two areas in which improvements could be made.

First, as the name suggested, keyword-based marketing reacts to the occurrence of a set of keywords rather than the content, where the keyword appear. Therefore, it is possible that an angry blog post containing the key- word causes an advertisement appearing. This is despite the intuition that both the reader and poster probably feel negative towards the mentioned product/service. To avoid wasting marketing budget, advertisement agencies used 'click-through' to charge a business only if the advertisement was clicked by a user. In the absence of a click-through, this means a wasted opportunity for an advertisement that could otherwise result in another sale.

Second, the keyword approach is highly sensitive to search results and the content that a user chooses to view.In that sense, a click-through is often "locally optimum".This approach can be seen as a technique rooted in traditional media. The idea is that higher viewership translates to higher conversion rates, i.e., click-through, even if the conversion rate is low, or the same. We argue that this assumption fails to exploit the characteristics of the digital world, where various mechanisms exist beyond viewership. For example, if real-time tools exist to help determine destinations for positive conversions, then campaigners can quickly shift destinations and target very specific pockets of content on a given Website.

As user-generated content continues to increase, we will see an eco-system of social sensors. These social sensors will be the force that shifts online marketing strategy, which is already happening. "Social sensors" is a concept referring to sources of information generated from within a community.While the information of each individual in the community does not have a significant impact,

their collective opinion, activities, responses, etc., are increasingly important to businesses. In this paper, we consider how such a paradigm shift may play out, and ask how the current keyword-based marketing model could be improved.

Our result is the development of a system that we call 'Cerebro'. Through Cerebro, a business could register for opportunities that brings a campaign. So rather than to bid for keywords through an "educated guess", Cerebro works by asking a business under what conditions it is looking for in pockets of activity.When such a condition happens, the business is informed and it could then make arrangements to hold a marketing campaign if it is chose. A condition on a pocket of activity includes keywords, sentiment, opportunity windows, social sources, and other factors. By allowing a condition to be established, Cerebro is able to identify more specific destinations than keyword-based search results.

The remaining sections of this paper will elaborate on our proposed system: we present the design details of Cerebro in the next section. In section 3, we will discuss the implementation details from the user's perspective and suggest how the design will translate to a scalable setting on a cloud platform. We then present the related works in section 4. A conclusion with a discussion of our future extensions to Cerebro is in section.

2. Cerebroas a Software Service

In brief, Cerebro is a software service requiring large amount of computing resources. It identifies activities such as blogs, conversations, comments, photo and video posts in the social sphere. The information generated out of these activities is the social voices and the mechanism that yield them called the Cerebro social sensors. Through techniques such as sentiment analysis, language processing and data mining, Cerebro identifies activities that match conditions requested by a business so as to enable them to target relevant destinations in real-time. In the following sub-sections Cerebro framework and its key components are presented.

2.1 General Architecture

The current version of Cerebro takes into account the following factors: (i) sentiment of a conversation, (ii) temporal relevance of a conversation, and (iii) matching net negative or net positive conversations to keywords registered by a business.

Figure 1 shows the architecture of Cerebro. The Social Sensors are Cerebro modules acting as proxies to social sites or social sources. For many sites such as FaceBook, YouTube or Twitter, the operators provide SDKs to access information generated by the users in real-time.In such cases, the Cerebro modules implements the SDK and APIs to achieve active monitoring of information generated. For such modules, the challenge of course is to find ways of adapting the individual SDK implementations to fit with the operating model of a social sensor in Cerebro. Without going into the details, the intent of each social sensor is to operate within the load requirements of each site while maintaining active 'crawling' of user- generated texts (e.g., comments). At this point, the current implementation is a straightforward 'crawl' of each social site. Illustrated the functionality of Cerebro, we expect future implementations to optimize its social monitoring according to the registered marketing opportunities. For other sites that lack official access mechanisms, wrappers [3, 4] are developed to yield information of interest to Cerebro. In most cases, wrappers areconsidered a last resort due to its sensitivity to structural changes on a social site.

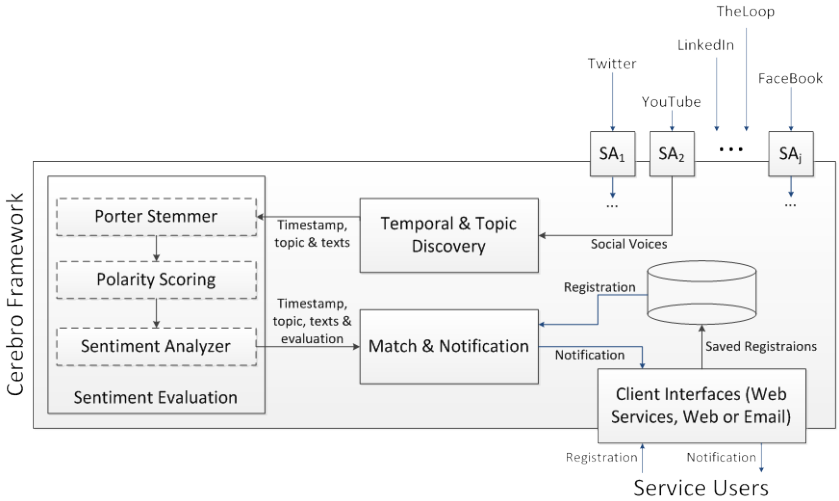

Figure 1:Cerebro Architecture.

A social sensor is a plug-in module in Cerebro, where its role is to activity monitor for social information.To give an insight into how this is currently achieved, we present the YouTube module because it is one of the first social sensor module developed in Cerebro. For this social sensor, the Google Data SDK was used to enable precise extraction of video comments. A seed list of users was used to kick start the process of monitoring. For each user, the list of videos uploaded and the most recent t comments can be downloaded.Since each comment on YouTube provides information about its poster, new users can be discovered to identify new videos and new comments consequently. This social sensor constantly runs to extract recent video comments to the "Temporal & Topic Discovery" module for further analysis by the evaluation engine. Thus, the concept at this point is similar to a crawler but eventually the prototype will be enhanced by using information in registrations, which achieves smarter monitoring behavior.

The three modules: "Temporal & Topic Discovery", "Sentiment Evaluation" and "Match & Notification" are a constantly evolving set of analytical facility. They work together to identify various user generated content that obtained through the social sensors. A series of related tweets for example is analysed for sentiment and produces placement opportunities, and then matched to registered interests. For each matching interest, Cerebro will notify its registrants of the destinations for the digital marketing opportunity.This registration of interests and the notification protocol is handled through the"Match & Notification" module and the "Web Interfaces" module.

Finally, the "Web Interfaces" of Cerebro provides the interaction point for users of the service to register their advertising interests. In the current prototype, a limited set of parameters can be specified along with each registration in Cerebro. This includes specifying the freshness of a group of social voices, the keywords and matching product categories of interests, the positivity or negativity of the social voices, and the method of notification for a registration.

2.2 Temporal & Topic Discovery (of Social Voices)

With YouTube, a major source of social voices is via the comments made in response to a video. While conceding that a video itself could be a source of social opinion, we hypothesize that the responses to an individual voice is more indicative of digital marketing opportunities. Thus, we shall only focus on video comments extracted by this social sensor. In a typical video response, comments are made over a period of time and the posts can be made asynchronously by multiple users. The social sensor thus extracts these comments along with their temporal properties forming a stream of short *texts*.

The simple approach of finding grouping of social voices from the texts can be done in two ways: one is to group them based on the similarity of their content. In this approach, we tend to identify the topic of each text but not the temporal relevance of all texts in a given timeframe. The other approach of clustering text is based on its temporal property risks grouping texts that have been interleaved with different topics. Either way, these two simple approaches do not achieve the first step of processing we need, which is to find topics of conversation (a series of related texts) in a given timeframe. A hybrid approach is thus needed and the algorithm needs to be designed with the ability to handle the incoming texts as a continuous stream.In other words, multiple scans such as first identifying related text and then ordering them according to their time property may not be adequate. Hence, during the course of designing Cerebro, we considered techniques such as k-Means [10], QBCA [9] and information-theoretic meta-clustering [11]. We also investigated the possibilities of developing their hybrids and that the results do not match up and/or that found the run time performance of those hybrids were poor. This motivated us to consider alternatives that a different technique can be developed to deal with this situation.

For the broad number of social sensors considered, we found social voices similar in structure and characteristics, i.e., they are often a stream of short texts.These texts can be video comments in the case of a social sensor like YouTube, or a tweet for a social sensor in Twitter. It is worthy adding to develop a specific technique for continuous stream of short texts ordered by their temporal properties and with topics interleaved within. The novelty of our approach is the use of a summarization measurement to achieve topic identification and summary in a single pass. The idea is based on two well-known measurements:lexical similarity [12] and text prestige [13]. Lexical similarity can be thought of as finding similarities between two sentences based on the overlapping use of overlap terms. So two sentences such as "Congress expected to vote on debt ceiling" and"Debt ceiling vote by Congress tomorrow" would be considered lexically more similar than either of the sentences against a sentence like "Barack Obama is the president of USA".

Our approach is to use the concepts introduced [12] in order to assess the lexical centrality of texts for a given window size. Due to all texts in the window are lexically similar, the prestige measure is calculated. The beauty of this approach is the ease in which both measures can be computed in one pass allows topics not only to be identified but also at the same time achieve topic summarization by picking the representative text in the topic. The temporal stamp of this representative text can then be used as the timestamp for evaluating currency of a topic. This one-pass approach means computational efficiency. That is crucial in processing the large amount of information extracted by each social sensor. Since the prestige measure will change as lexically similar sentences appear in the window, a current topic is updated along with each evaluation. This means that a topic is no longer active will be removed naturally in the process as the stream moves through the window.The window size thus determines the number of active topics that is tracked.A bigger window will allow more topics to be tracked and will keep a topic alive for a longer period of time.

2.3 Sentiment Evaluation

Given a stream of texts from a social sensor, the "Temporal & Topic Discover" module identifies a topic from collecting texts, selecting a time stamp and defining set of representative texts for each topic. This output is then channelled to the "Sentiment Evaluation" module to determine the mood (or sentiment) of that topic. There are many techniques to determine the sentiment of a given document, containing a sequence of texts.Sentiment can be evaluated by using lexical resources, natural language processing, or machine learning algorithms. Depending on the characteristics of texts, some techniques work while others don't.

In considering the design of this module, our stream of text are social voices with a wide range of topics. Hence, techniques such as machine learning is infeasible due to the amount of human experts who are required to prepare and provide the training sets. What would be more appropriate in our case is a more generic approach that is based on lexicon resources and natural language pro- cessing. Our implementation in Cerebro is based on SentiWordNet [1], a lexical resource for evaluating an opinion that is based on three parameters: positivity, negativity and objectivity. These three parameters form is called a "polarity score" and is associated with a term in WordNet [14]. The parameters of a polarity score will always add up to one so that a term 'horrible' will have a score expressed as a synset, i.e., 0 (positivity,$\mathcal{P}$), 1 (negativity, $\mathcal{N}$), 0 (objectivity, $\mathcal{O}$). Likewise, a term such as 'fantastic' will have a synset1, 0, 0 giving a polarity score of 1 as well. The polarity scores of a term is automatically created through a combination of linguistic and statistics classifiers. The current version used in Cerebro, SentiWordNet 3.0, uses WordNet 3.0 and includes additional mechanisms to refine the scores, including the use of semi-supervised learning and a random-walk by constructing WordNet 3.0 as a graph. So far, the results reported in the literature suggests that SentiWordNet is promising for generic sentiment evaluation. Hence, we choice it for building this initial version of Cerebro.

As shown in Figure 1, the "Sentiment Evaluation" module is made up of 3 sub-components (for now).To effectively evaluate the sentiment of a text, each term has to be first stemmed. This means that a word term in English can have many variations, e.g., 'run' vs. 'Running' but carries same semantic. Stemming is a process to convert the variations into their 'base' form. This effectively increases the matches of a term in SentiWordNet; thus, the accuracy of the overall polarity score of a sentence is improved. For example, a social voice in response to a Photoshop video was "very nice, good explanation for every step. Thank you for an extremely helpful tutorial.".After the Porter Stemming algorithm [5], we have "nice, good explain step.thank,extreme help tutor". This stemmed text is used in the evaluation of a polarity score.

The evaluation of a polarity score of a text is rather straightforward. First the text is stemmed. This removes 'stop words' and reduces the word terms to their 'base' form. Then for each stemmed word term, we look up the SentiWordNet database to obtain its individual polarity score. For some word terms, a single synset exists for others, and multiple synsets appears in the SentiWordNet database. In this case, the term 'nice' has six synsets; hence, the polarity score of the term is the average of the synsets, i.e., for a word term that could be an adjective, verb or noun, we collect each synset and then derive the final score by averaging each component in the synset:

$$PS(\omega) = \left\{\frac{1}{n}\times\sum_{i=1}^{n}\mathcal{P}(\omega,i), \frac{1}{n}\times\sum_{i=1}^{n}\mathcal{N}(\omega,i), \frac{1}{n}\times\sum_{i=1}^{n}\mathcal{O}(\omega,i)\right\} \quad (1)$$

where $PS(\omega)$ is the polarity score of a word term ω. To derive the polarity score of a text sentence, the same evaluation technique is applied over all word terms in the sentence, i.e., by finding the average of each component in the synset over all word terms.

This sentence level polarity score is fed through the sentiment analyzer along with the original texts to collate with texts extracted by other social sensors. Since multiple voices on the same topic and sentiment can co-exist across the different social sources, the role of the sentiment analyzer is to catalogue related group of texts based on the topic and then index them according to their sentiment

values. This information is used by the "Match & Notification" module. This is discussed in next section.

2.4 Match & Notification

The "Match & Notification" module is rather straightforward in the implementation. A topic is essentially unlabelled but identified by the "Temporal& Topic Discovery" module through a sliding window and the lexical centrality of text. This grouping of related voices with an unlabelled topic allows us to by-pass machine learning training and do away with human expertise in the process. That allows Cerebro to monitor on a large scale and across the social sphere. Without labeled topics associated with a grouping of texts representing social voices, a different approach is required to identify topics that in turn uncover the sentiment for targeted digital marketing opportunities.This is done similar in concept as current keyword-based marketing with a small twist.

In the case of Cerebro's keyword implementation, users can identify a group of keywords that would arise in a given topic.These keywords are then used to match a conversation (i.e., a group of texts) allowing us to do away with a topic. This approach has the added advantage of multi-labeling the same conversation for different purposes and users. Unlike keyword-based marketing, this module is more than matching registered keywords in a conversation. The 'matching' aspect of this module actually uses an ontological database so as to allow a keyword to reach more voices even if a keyword is not directly presented. For example, a conversation about 'sunglasses' can be matched to 'shades' (the keyword registered by a user) without any human intervention. This is achieved through a process of product ontology matching. In the current prototype, the ontology matching is achieved through a product category tree developed by Google (see **http://www.google.com/basepages/producttype/taxonomy.en-US.txt**).

We converted the tree into a data structure internal to the "Match & No- deification" module and then run each terms through a synonyms database. The terms are manually updated to consider multiple word product terms. While we have yet worked on a matching mechanism allowing multiple terms in a text correctly.This is definitely one of the future work that we will undertake as an upgrade of Cerebro's implementation. Currently, the prototype matches on registrations to actual keywords as well as any single term synonym matches. For all these matches, notifications are sent.

A user can choose a number of notification options, either (i) in program through a Web Service call back or (ii) by email as a report to advice of digital marketing opportunities based on the parameters specified. The first allows automation at the user's end when the later allows expert evaluation and decision making.

3. Implementation

The prototype was developed by using .NET and C#, utilizing relevant .NET libraries include the Google Data SDK (**http://googledataapi.codeplex.com/**), SentiWordNet 3.0 [1], and the Porter stemming algorithm [5]. In addition, a synonym data structure was created from the Google Product Taxonomy for Cerebro to identify product placement opportunities. The system is exposed to the users mainly through a Web Service interface for automation and a Web-based user interface for manual control. The next section briefly discusses the current Web Service interface implementation.

3.1 Programming Interface

Cerebro's programming interface is based on XML Web Services. It allows third-party implementations without confining to the .NET environment.From the development perspective, Cerebro is available to the user as a Web Service class in the **DeakinITB.ServiceComputing** namespace. Currently, the Cerebro class has only a few publicly available methods. The first is to enable registration of monitoring requests. This is defined in the Cerebro interface as below:

```
stringCerebro.Register(string Settings);
```

where Settings is an XML formatted document containing the monitoring setup. The detailed information required can be seen via the Web interface. In brief, they include fields as show in Figure 2. Much of them in Figure 2 has been explained in the caption except the token attribution of the register tag, which is obtained when a user applied for an account with Cerebro. For a registration to be lodge, the token must be supplied to identify its owner. If the registration is successful, the returned XML string will contain an ID. This ID is used for future references to the registered request. On the other hand if the call fails in anyway, the returned string will hold the error information. The **Register()** method is the most importantinterface between the user and Cerebro. Other interfaces are mainly for maintening of user requests, such as the update of a registration or to remove one.

3.2 Cloud Architecture

For this system to deliver its promise, large volume of data has to be extracted from social sites and then processed to match a given registration. To give context to the volume and scale, there are about 53 socials sites carrying large amount of social voices [2]. On top of that, there are further millions of active blogs all over the world, which are also sources of social voices. Hence, not only must Cerebro be capable of handling huge volumes of data, but also handle large number of registrations and notifications. The computational power and service availability requirements mean that Cerebro must be executed on a highly scalable and redundant architecture.

On the basis of such possible load, we considered the available options and were attracted to the cloud computing paradigm. This is because Cere bro modules can be deployed on separate virtual machines, replicated for redundancy and scalability, and supports communication efficiently through fast connect networks within the cloud. Due to the chance of a server failure we also considered the characteristics of cloud applications such as the need for stateless-ness. For most of parts, Cerebro's modules are stateless with the exception of the social adaptors.They need to keep track of their states with regards to the progress of their crawl.The other exception is with the possibility of a failure in the midst of a registration and in the midst of performing a notification. Most of the other modules, their execution is pretty much stateless and on-going. Consequently, this makes it easy to scale Cerebro over multiple machines dealing with the required load.

An interesting design aspect of Cerebro is that there isn't an explicit persistent storage for the data that obtained from the social sensors. Instead, data are sent through to the "Temporal & Topic Discover" module, this means the module is responsible for an inherent temporary storage to process the social voices. This inherent temporary storage could be a large memory space running in the same machine as the module, or on a database that may not be on the same machine as the module in a cloud setting. We suggest that it is best to avoid explicit storage in the design in order to allow flexibility in the implementation on different types of a platforms. In a cloud setting, better performance may be achieved through memory rather than a networked database.

4. Related Works

To our knowledge, Cerebro is the first generic monitoring mechanism for social voices to improve digital marketing placement opportunities. Closest to Cerebro is a project called **BrandKarma.com** that also collates social voices of a brand and maps out the 'hot' topics discussed. The similarity between Cerebro and **BrandKarma.com** lie in their use of social voices, one is meant for community evaluation of a given brand. The other is to improve click-through or sales outcomes. Although it is possible for a company to monitor social voices through **BrandKarma.com**, the process will be only manual and pinpoints in the active discussions. Cerebro, on the other hand, provides the automation through a registration process, and reverts information on those where active discussions take place, which are based on the given monitoring criteria.

Other works with Cerebro are largely related to fundamental re-search rather than the applied research as this study. In one area, there is a body of work on analysing product reviews the commonly found on forums. For example in [15], a model based on the inter-relationship of 'helpfulness of product reviews' on **Amazon.com** was developed to gain objective insights into customer opinions. Variations of analysis on product/service reviews in this area were studied across different domains using different approaches.Some representative works include [16, 17, 20], etc.

In addition to product reviews, another related area is on the detection of sentiment.This area of work is sometimes known by other names such as opinion mining or sentiment classification. In addition to SentiWordNet [1] used in Cerebro, other works in the area of sentiment evaluation including [18, 19]. In most cases, the main body of work focused on detecting sentiment from a body of texts rather than short social voices. This is why we made a decision of using SentiWordNet against the other approaches, owing to SentiWordNet provided the basic mechanism to evaluate sentiment from short texts rather than a document. The only other work that evaluates the sentiment of short texts is SentiStrength [21]. Rather than making an exclusive choice between SentiWordNet and SentiStrength, our intent is to develop an ensemble model out of them as a future improvement work of Cerebro. We favor this approachas our literature review in the area of data analysis. The literature suggests that ensemble approaches tend providing better overall results when set in a generic context.

5. Summary

This being the first initial prototype of a system certainly has many areas for improvement. More than a prototype, indeed, Cerebro underpins a frame- work for our investigation of novel computing applications that uses social media data and requires the compute power of a cloud. With large amount of social media data been generated constantly through mobile devices such as smart phones and tablets, the ability of analyzing the collated continuous stream of social information could only be achieved on a powerful computingplatform such as the cloud. To justify such costs and huge amount of effort, it exposes a software service on paying per use model makes the most sense. Cerebro is such a framework and the currently undertaking work is a step to start moving towards such a possibility.

It is also important to note that Cerebro is not a replacement of current digital marketing models such as Google's AdWords.Rather, it is a service to complement current digital marketing techniques. Cerebro has chieved success in improving the effectiveness of digital marketing outcomes for advertisers by pro-actively monitoring and reporting to the advertisers on where they could potentially direct their resources to achieve the best digital marketing outcomes. From the applications perspective, Cerebro is novel as a technology that is on the side of advertisers. With a research perspective, it is an interesting framework for various applied research activities that realize the potential of cloud computing and social network services.

Indeed a number of future works are already on-going that is just because of the current version of Cerebro. A framework to support open plug-ins of social sensors is first on our list. Our premise is to create an open platform so that a large number of social sensors can be quickly developed and allow new social platforms providing the data to Cerebro to easily create such a channel. This approach would allow new data being quickly captured than internally developing only a few of our coders within the group.

The "Temporal & Topic Discover" module is currently designed only for handling a stream of social voices from a single source. Eventually, multiple streams from different social sensors need to be collated for analysis. How this can be done in the context of a concurrent cloudenvironment is an issue to be investigated. The "Sentiment Evaluation" module is definitely a Cerebro component that needs a constant improvement. The premise of Cerebro depends on the accuracy of the "Sentiment Evaluation" module and a constant fine-tuning is expected. Our immediate update on this module is to consider the result report in [8] as way to improve the sentiment assessment for each identified conversation. The other conducting experiment is the ensemble sentiment evaluation technique mentioned earlier through the use of SentiStrength [21]. Finally, the "Match & Notification" module also requires more advanced handling of ontological matches to improve the accuracy of notifications. Currently, matching is only a single term leading unwanted notifications. Further work to match concepts beyond a single word term will be considered. As we improve on Cerebro's features and work on refining the system to migrate Cerebro onto commercial cloud platforms, there will be another updateprovided.

References

[1] Andrea Esuli, Stefano Baccianella and FabrizioSebastiani. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining.Proc. 7th Conf. International Language Resources and Evaluation. European Language Resources Association, Valletta, Malta, 2010.

[2] Kevin Chai, VidyasagarPotdar, and Elizabeth Chang. A Survey of Rev- enue Models for Current Generation Social Software's Systems. Proc. Int. Conf. on Computational Science and its Applications, Osvaldo Ger- vasiand Marina L. Gavrilova (Eds.), Vol. PartIII. Springer-Verlag, Berlin, Heidelberg, pp. 724-738, 2007.

[3] Yue-Shan Chang. Adaptable Wrapper Generation for Web Page Format Change. Proc. 5th Int.Conf. on Applied Computer Science, Wenhao Huang, Z. Y. Hu, Q. Z. Chen, and S. Y. Chen (Eds.). World Scientific and Engineering Academy and Society, Stevens Point, Wisconsin, USA, pp. 147-152, 2006.

[4] Juan Raposo, Alberto Pan, Manuel Alvarez, and Justo Hidalgo. Automatically Maintaining Wrappers for Semi-Structured Web Sources. Data Knowledge Engineering, 61(2), pp. 331-358, May 2007.

[5] M.F. Porter. An algorithm for suffix stripping, Karen Sparck Jones and Peter Willet (Eds.). Readings in Information Retrieval, San Francisco: Morgan Kaufmann, ISBN 1-55860-454-4, 1997.

[6] Huifeng Tang, Songbo Tan, and Xueqi Cheng. A Survey on SentimentDetection of Reviews. Expert Systems with Applications. 36(7), pp.10760-10773, September 2009.

[7] Chenghua Lin, Yulan He, and Richard Everson. A Comparative Study of Bayesian Models for Unsupervised Sentiment Detection. Proc. 14th Conf. on Computational Natural Language Learning. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 144-152, 2010.

[8] Yue Lu, MaluCastellanos, UmeshwarDayal, and ChengXiangZhai.Automatic Construction of a Context-Aware Sentiment Lexicon:an Optimization Approach. Proc. 20th Int.Conf. on World Wide Web. ACM, New York, NY, USA, pp. 347-356, 2011.

[9] Zhiwen Yu and Hau-San Wong. Quantization-based Clustering Algorithm. Pattern Recognition, 43(8), pp. 2698-2711, August 2010.

[10] Ming-Chao Chiang, Chun-Wei Tsai,and Chu-Sing Yang. ATime- Efficient Pattern Reduction Algorithm for k-Means Clustering. Information Science, 181(4), pp. 716-731, February 2011.

[11] Shin Ando and Einoshin Suzuki. Detection of Unique Temporal Segments by Information Theoretic Meta-Clustering. Proc. 15th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. ACM, New York, NY, USA, pp. 59-68, 2009.

[12] GunesErkan and Dragomir R. Radev. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. J. Artificial Intelligence Research, 22(1), pp. 457-479, December 2004.

[13] Michael J. Albers. Information Salience and Interpreting Information.Proc. 25th ACM Int. Conf. on Design of Communication. ACM, NewYork, NY, USA, pp. 80-86, 2007.

[14] VeronikaVincze, GyorgySzarvas, and Janos Csirik. Why are WordNets Important? Proc. 2nd Conf. on European Computing, CostinCepisca, Guennadi A. Kouzaev, and Nikos E. Mastorakis (Eds.). World Scientific and Engineering Academy and Society, Stevens Point, Wisconsin, USA, pp. 316-322, 2008.

[15] CristianDanescu-Niculescu-Mizil, GueorgiKossinets, Jon Kleinberg, and Lillian Lee. How Opinions Are Received by Online Communities: A Case Study on Amazon.com Helpfulness Votes. Proc. Int.Conf. on World Wide web. ACM, New York, NY, USA, pp. 141-150, 2009.

[16] F. Maxwell Harper, Daphne Raban, SheizafRafaeli, and Joseph A.Konstan. Predictors of Answer Quality in Online Q&A Sites. Proc.26th Annual SIGCHI Conf. on Human Factors in Computing Systems. ACM, New York, NY, USA, pp. 865-874, 2008.

[17] Yue Lu, ChengXiangZhai, and Neel Sundaresan. Rated Aspect Sum- marization of Short Comments. Proc. 18th Int. Conf. on World Wide Web. ACM, New York, NY, USA, pp. 131-140, 2009.

[18] Bo Pang, Lillian Lee, and ShivakumarVaithyanathan. Thumbs up? SentimentClassification Using Machine Learning Techniques.Proc. Conf. on Empirical Methods in Natural Language Processing, Vol. 10. Association for Computational Linguistics, Stroudsburg, PA, USA, pp.79-86, 2002.

[19] Matthew Richardson, AmitPrakash, and Eric Brill. Beyond PageRank: Machine Learning for Static Ranking. Proc. Int. conf. on World Wide Web. ACM, New York, NY, USA, pp. 707-715, 2006.

[20] Fang Wu and Bernardo A. Huberman. How Public Opinion Forms.Proc. Int. Workshop on Internet and Network Economics, Christos Papadimitriou and Shuzhong Zhang (Eds.). Springer-Verlag, Berlin, Heidelberg, pp. 334-341, 2008.

[21] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai and A. Kappas. Sentiment Strength Detection in Short Informal Text. Journal of the American Society for Information Science and Technology, 61(12), pp. 2544-2558,2010.