"Vasile Alecsandri" University of Bacău
Faculty of Sciences
Scientific Studies and Research
Series Mathematics and Informatics
Vol. 19 (2009), No. 2, 163 - 178

KNOWLEDGE MINING FROM WEB CUSTOMER OPINIONS TO IMPROVE ENTERPRISE PRODUCT

DOMENICO CONSOLI, CLAUDIA DIAMANTINI, DOMENICO POTENA

Abstract. Nowadays, for the enterprise, the customer is a very important strategic resource. The opinions of a customer about a particular product/service helps top management to introduce improvements in processes and products, thus the enterprise gains competitive advantages. So, it is very important to define a bidirectional communication channel, supported from collaborative tools, between the customer and the enterprise. In this paper we introduce a customer oriented framework that after gathering and polarizing web customer opinions, with an algorithm of sentiment analysis, is able to route dysfunctions about product/service to competence center. This framework can be considered a Customer-Centred Information System that crosses many internal business functions.

1. Introduction

Customer opinions constitute a gold resource for making strategic decisions [1]. Nowadays in the web sites, forums, chats, blogs as epinions.com, cnet.com, complaints.com,... there is an huge amount of opinion freely express from customers [2]. Customers prefer to express their opinions by using natural language [3], interview by phone, face to face or free writing rather that answering to a structured or pre-structured questionnaire.

Keywords and phrases: ultrametric space, spherically complete, fixed point, implicit relation.
(2000)Mathematics Subject Classification: 47H10, 54H25

Customers exploit web 2.0 tools for both expressing their opinions about a product and suggesting solutions for improving it. The enterprises often encourage exchange of opinions, by making available virtual communities, e.g. Italian Nikon's Camera forum, where people review Nikon products (http://www.nital.it/forum/), the blog on Benetton products
(http://benettontalk.com), and so on.
To capitalize customers opinions is very important for an enterprise, to project customer-centric Enterprise Information System to give, in real-time, the situation of the market.

The *Customer enterprise Customer (CeC)* model [4], presented in this paper, continuously hears customer opinions about the product/service and behaves accordingly. The model encourages the creation of a a very important bidirectional communication channel between customer and enterprise. The customer plays an active role in the enterprise, in particular he/she participates as a co-producer (i.e.*IKEA*) or as a consultant(i.e.*500 Wants You - www.fiat500.com*).
CeC first collects and analyses web customer opinions freely available on the web sites, then reports dysfunctions about the product/service to competent offices for the necessary improvements or for answering customers.
This paper is organized as follows: in the next section we give a brief description of CeC model. In third section we show a Sensing phase. The most important Mapping module is presented in section fourth with relative algorithm and case study. In the fifth section we present Actuation phase. Finally some conclusions are drawn.

2. The Customer enterprise Customer Model

In Fig. 1, the Customer - enterprise - Customer (CeC) model is shown.

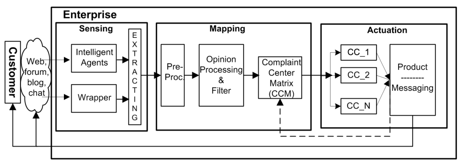

Figure 1. The Customer-enterprise-Customer model

The model is divided into three phases: *Sensing*, *Mapping* and *Actuation.*
Sensing scans the web for finding and gathering customer opinions. Then, opinion is processed by a sentiment analysis algorithm and filtered to extract only useful complaints (negative opinions).
In the mapping phase, complaints are mapped and routed to specific competence centers (CCs), that can be a center or a groups of competent people.
The goal of the actuation phase is to react to external stimuli, either by making changes to the product or by answering to customers (messaging).

3. Sensing

The sensing module is composed by a set of crawler agents or wrappers. Crawler agents are specialized in different protocols (http, https, pop3, imap4, nntp) , that are responsible to inspect and retrieve information, respectively, from web sites, blogs, chats, e-mails, newsgroups and so on. Each agent can be configured with policies to extract web text with advanced techniques of Natural Language Processing (NLP). Alternatively, users may, apriori, select web sites of interest and define wrappers in order to analyze the structure of each site and extract opinions. In the literature there are various examples of wrappers [5].

4. Mapping

This phase is the more important of model. We can distinguish three subsection: - Preprocessing - Opinion processing - Complaint Center Matrix

4.1. **Preprocessing.** Since opinions are written in Natural Language, to process them, we need specific pre-processing techniques and in particular to this end we have used the General Architecture for Text Engineering (GATE) library. The goal of this phase is to obtain for each opinion, expresses in a web post, statements and significant words. The preprocessing consists of the following steps: sentence extraction, statement extraction, tokenization, stemming, lemmatization, part of speech and elimination of stop words. From every web post, eliminating all interrogative clauses, we extract minimum sentences. After we divide the sentence in statements. A statement is an elementary subsentence that expresses a single positive, neutral or negative polarity

while a single sentence can express more than one opinion. For dividing sentences in statements we separate the sentence in the proximity of those conjunction that link two propositions with opposition affectivity; for example "but" (coordinative conjunction) or "although, even, thus, whereas, while" (subordinate conjunction). After we divide statement in single token or word. With stemming and lemmatization we derive the root of words, removing affixes and endings. In the Part of Speech phase every word of a statement is labeled by a tag with the correct part of speech: article, noun, verb, adjective, etc.
After pre-processing phase we obtain a statements-words matrix W (n x m) where the generic element w_{ij} represents the frequency or number of occurrence of a word j in a statement i, with $i = 1, \ldots, n$ and $j = 1, \ldots, m$.

4.2. **Opinion processing.**

4.2.1. *Our proposal and state of art.* For polarizing statements we use an original algorithm of sentiment analysis that mainly focuses on six Ekman emotional indexes [6]: happiness, surprise, fear, sadness, anger, disgust. Paul Ekman , in the study of facial expressions and emotions, noticed that all human beings respond with the same facial movements to the same emotional states that derive, from the main six emotions.
In the same way, in our algorithm, we think that each word of text expresses sentiments and emotions and there is a connection with sentiments, emotions, affectivity present in the text and a polarity of customer opinions. The consumer shift its purchase behavior from needs to emotions/experiences. For example, in luxury goods, the emotional aspects as brand, uniqueness and prestige for purchasing decisions, are more important than rational aspects such as technical, functional or price. An other factor that influence purchasing customer is, for example, the disgust. Customer don't buy disgusting products. The disgust is a repugnance toward any object, action or person.
Other authors of sentiment analysis algorithms focused only on single words. Authors that use Ortony model [7] consider three main classes of emotions: satisfied/unsatisfied, approved/disapproved, pleasant/unpleasant.
Esuli and Sebastiani [8] have created SentiWordNet, a lexical resource

for opinion mining, where they assign to each synset (set of synonyms) of WordNet sentiment scores: positivity, negativity and objectivity (i.e. neutral). WordNet Affect [9] is a linguistic resource for a lexical representation of affective knowledge. In WordNet Affect each synset of WordNet is labeled by one or more affective-labels (emotions, mood, cognitive or physical state), representing the affective meaning of the synset.

4.2.2. *Affective polarity model.* The goal of this phase is to calculate affectivity of IAW and IAS and its polarity from the manual assignment of DAW and DAS.
In the following, it is described the algorithm for building the affective polarity model:

Manual assignment. In our model we classify words of statements in: Direct Affective Word(DAW) and Indirect Affective Word(IAW). The DAW group is formed by words expressing an emotional state in the specific domain. For example, the words *happiness* and *cheerful* carry a positive emotional state. These words, that contain minor errors, are independent from the context and convey always the same emotional state.
Words that directly don't express any emotional state but, in combination with DAWs present an affective value in a statement, belong to IAW group. For example, the word *ice* by itself doesn't convey any emotional state. If we insert in the word *I like ice* the word acquires a positive emotional state clearly different from *I hate the ice*.
Respect to statements we can make the same distinction: Direct Affective Statement (DAS) and Indirect Affective Statement (IAS).
We annotate manually only DAW and DAS associating a vector of six Ekman emotional elements (happy, surprise, fear, sad, angry, disgust).
The value of each component can assume a value between 0 and 10. For example, if we associate to the word "stench" the affective vector (0, 0, 0, 2, 2, 6), it means that in the definition of the affective meaning of the word stench, the elements sad and angry contribute with a small value, the element disgust with a high value and other elements don't produce any contribution.

Word affectivity. The value of new affective vectors of IAW depends on the affective vector of the most similar DAW in the statements-words space. The idea is that similar words transport also similar affective state.

The computation therefore bases on similarity concept evaluated by the normalized scalar product k

$$k = \frac{w_p.w_q}{\|w_p\|.\|w_q\|}$$

where w_p, w_q are affective vectors of words p and q, $w_p \in IAW$ and $w_q \in DAW$ and $w_p \cdot w_q$ is the scalar product of two vectors.

For calculating new affective vector Aw_p [10] of word w_p of IAW we use two methods:

$$(A1)\, Aw_p = k \cdot Aw_q \quad k > s \quad \forall w_p \in IAW \; \exists\, w_q \in DAW$$

$$(A2)\, Aw_p = Aw_q \quad k > s \quad \forall w_p \in IAW \; \exists\, w_q \in DAW$$

In the last case we consider that affective vector Aw_p is egual to similar vector Aw_q of DAW set .

In both cases s represents a threshold. Increasing this threshold value affective vectors are calculated only for words very similar to the DAW. Words, with their affective vectors, whose k is below a threshold s, are not considered.

Statement affectivity. The method that we use to calculate the affective vector As_i of statement i is the following:

$$As_i = \frac{\sum_X DAW}{n_{DAW}} \cdot \alpha + \frac{\sum_X IAW}{n_{IAW}} \cdot (1 - \alpha)$$

$$\forall s_i \in IAS \quad \alpha \in [0, 1]$$

where X is the set of direct affective words of statement S_i while Y is the set of indirect affective words. The parameter α indicate the weight of DAW on estimation of statement affectivity. In this estimation (1- α) is the weight of IAW.n_{DAW}, n_{IAW} are respectively number of words directly and indirectly affective.

Polarity estimation. We propose two methods to estimate polarity p of a statement that derive from affective vectors. The first method focuses on affective value of surprise since this term may assume positive or negative meaning depending on predominant sentiment in the statement (difference between "happy" and negative indexes). The second method uses a linear regression for polarity estimation.

Method SP1. Surprise index follows prevalent sentiment of statement.

$$p = \frac{1}{2}\sum_{i=1}^{2} e_i \cdot \alpha_p - \frac{1}{4}\sum_{i=3}^{6} e_i \cdot (1-\alpha_p)$$

if happy $\geq$ (sad+angry+fear+disgust), otherwise

$$p = e_1 \cdot \alpha_p - \frac{1}{5}\sum_{i=2}^{6} e_i \cdot (1-\alpha_p)$$

where $(e_1, e_2, ...e_6)$ are indexes of affective (emotional) vector ("happy", "surprise", "fear", ..."disgust"), α_p is the weight assigned to positive indexes, $(1-\alpha_p)$ correspond to α_n (weight assigned to negative indexes).

Method SP2. Estimating statement polarity by linear regression.

$$p = a + \sum_{j=1}^{6} a_j \cdot As_j + e$$

where a is intercept of straight line, a_j are regression coefficients of six affective indexes, As_j the six elements of statement affective vector and e the statistical error. There is a linear relationship between the values of statement affective vectors and its polarity. Regression allows us to calculate the coefficients a_j.

4.3. **Case study.** In order to test the validity of our methodology we have gathered 800 posts from web forums on customer opinions about a resort in Sharm el-Sheikh and in particular we selected opinions about services: Kitchen, Restaurant, Room Service, and Administration.

After the pre-processing phase, the software saved into database 1303 statements and 2300 words.

We labeled manually 900 DAS and 374 DAW. The value of affective index for each DAW or DAS varies between 0 and 10 (values controlled by the software). Zero means no affectivity while the value 10 express an highest amplitude of affectivity. For DAS a polarity value p varies between -10 and 10.

Experimental planning. In our experiments we consider a Training Set and a Test Set for words (TrainW and TestW) and for statements (TrainS and TestS). In both case we consider as training set the 50% of DAW and DAS manually assigned. Remaining 50% forms the test set. In this way we can compare manual assigned values with values calculated by software.
We vary TrainW and TrainS from 20% to 100%, threshold s from 0,1 to 0,5 and parameter α from 0 to 1. In total with all combinations we have made around 4000 experiments. In the affectivity estimation, as error we consider the mean squared error (MSE) and the standard deviation (σ) of it. For polarity estimation we consider the amplitude and the sign. The amplitude error is given from the normalized difference between assigned and estimation polarity $e = |p_a - p_e|$. The percentage sign error (misclassification) will be given by the sum of false positives and false negatives divided by the total statements of the test set.

Experimental results. The statement affectivity and its polarity depend on word affectivity. For estimating word affectivity we consider both methods with similarity coefficient k (A1) and without k (A2). As concerns the method A1 increasing the TrainW the error increases, because there is the a negative influence of K that transports an intrinsic error while with the method A2 increasing TrainW, error decreases.
Regarding the statement error to vary of TrainW it is important to comment the Figure 2. Initially as trainW increases the error decreases. For high values of TrainW, the error introduced by the coefficient k is high and therefore the statement error increases. To this point the threshold s is important. When the value of s is 0.5, the number of words contributing in the estimation of the statement affective vector is smaller and so reduces the MSE. Figure 3 shows the statement error to vary of α parameter. A low value of α means a more influence of IAWs and then the error is high. Increasing the value of α, the contribution of DAWs is greater than IAWs and the error decreases.

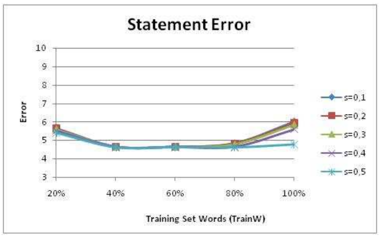

FIGURE 2. Statement error to vary of TrainW

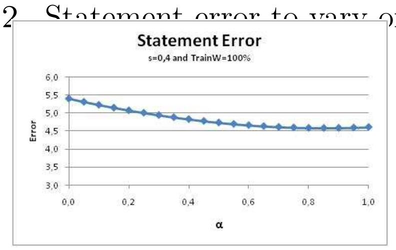

FIGURE 3. Statement error to vary of α

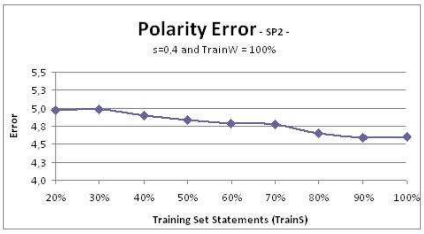

FIGURE 4. Polarity error

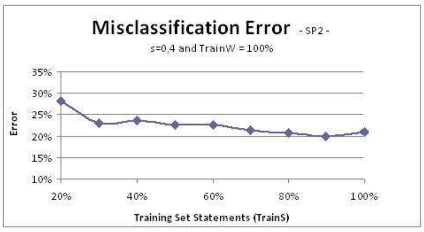

FIGURE 5. Misclassification error

Regarding polarity estimation with the increasing of TrainS, amplitude error (Figure 4) and misclassification error (Figure 5) decrease.

The polarity error decreases also increasing α_p parameter. Comparing these experiments we can say that better methods for word and statement affectivity is A2 method and SP2 (regression linear) for polarity estimation. The system of classification, for best configuration, may categorize words with an error of 21%, statement affectivity with a MSE of 14%, polarity amplitude with 22% and a misclassification error with 20%. These results are encouraging considering that we have used for our experiments a small training set size. The results can improve with a bigger training set, a manual labeling more rigid, with adjustments of K,s, α parameters and with the use of polynomial regression or neural networks in the polarity estimation.

4.4. **Filter.** The statements, after Opinion Processing phase, are filtered to select only complaints (negative opinions) to send competence center. In this phase positive opinions are eliminated.

4.5. **Complaint Center Matrix.** The Complaint Center Matrix (CCM), is a matrix that associate a complaint (negative statement) to one or more Competence Centers (CCs) (Table I).

Human resources (Competence Centers)

	Chef	Maitre	Director
The first dishes are bad	X		
The room service is terrible			X
Waiters are ungraceful		X	
Bed mattresses are hard			X
The beach is dirty			X
The meat is very salty	X		
....	......	......	......

TABLE 1. An example of Complaint Competence Matrix

In our project of tourist resort we consider three CCs: Kitchen, Restaurant, General Services that refer respectively to three competent human resources as Chef, Maitre Hotel, Director (Table II).
Mapping a statement to specific resource can be solved with algorithm of Text Mining and in particular with a Text Categorization or Text Classification [11]. Many authors use algorithms with a training set by examples, previous classified in known classes.
A training example is an instance $x \in D$ (domain), paired with its

Resource	**Skills**
Resort Director	- responsibility of the human resources - beach services - room services - ......
Maitre Hotel	- waiters responsibility - customer care - toasting drink at table - ......
Chef	- kitchen responsability - food preparation - fish and meat - ......

TABLE 2. Human Resources Profiling

correct category $c(x) : (x, c(x))$. For categorizing a text we must find a hypothesized categorization function, $h(x)$, such that:

$$\forall (x, c(x)) \ \in D : \ h(x) = c(x)$$

Text Classifier, trained, learns to recognize the characteristics of the categories of interest and so it is able to classify a new document. The systems of Text Classification use different approach [12] of knowledge engineering and learning algorithms as bayesian, neural network, nearest neighbor, support vector machines,....
In our complex project we are developing a statistical text classification methodology that interfaces with our sentiment analysis software (SAS). We assign a complaint, with associated affective vector, to responsible people. It is important that human resource responsible, primarily, understanding on what complaint focuses: the product/service is disgusting, the customer is angry, he has afraid of wrong choice and then changes supplier or he is sad because unsatisfied from the product. The value of statement affectivity is also important for following strategic moves: *Fear*(special promotional campaigns for closing the customer), *Anger*(reassure and accompany customer in post-sale paths), *Sadness*(gladden customer with unique gadgets), *Disgust*(improve immediately the product design).
Regarding the polarity it is important not only the sign + (satisfied) - (unsatisfied) but also the amplitude. The polarity amplitude indicate how much a customer is satisfied or unsatisfied. An high value of

amplitude indicate that enterprise must intervene with urgency and priority. For this reason in SAS we use statistical object integrated with affective objects that polarize the opinion.
In particular we use three different objects: TFIDF, OccurenceMatrix and StatementFormat.
TFIDF module is useful to calculate the parameter term frequency-inverse document frequency (tf-idf) [13]. The tf-idf is a statistical measure used to evaluate, with a weight, how a word is important in a document of corpus. The importance increases proportionally to the number of times a word appears in the document but is inversely proportional to number of times that the word appears in the entire corpus.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_{j=1}^{k} n_{i,j}}$$

where $n_{i,j}$ is the number of time that the word i appears in document j and the denominator represent the number of time that a word i appears in the corpus of k documents.
It is clear that if a word appears in a document and doesn't appear in other documents, that word is a characteristic, a feature of specific document.
Regarding other two objects, OccurrenceMatrix contains the matrix of occurrence of various words after preprocessing phase and StatementFormat the feature of a statement and in particular the six elements of statement affective vector (happy(h), surprise(s), fear(f), sad (sd), angry(a), disgust (d) and polarity (p)).
If we give in input to TFIDF module the StatementFormat and OccurrenceMatrix objects we obtain for each statement a vector of number, weighted in the corpus of domain, where each number indicates the most important words in the specific statement. For each statement the relative affective vector is saved in the StatementFormat object.
In detail our algorithm presents the following stages:

1. *Weighed numerical vector.* For example if we consider the statement "The swimming of resort α is big and beautiful" the module TFIDF return a sorted and weighted vector as (0.8(swimming), 0.7 (resort), 0.6 (beautiful), 0.2 (big), (0.1)(α)). In this example, if we consider a threshold of $s = 0.5$ the more significant words of statement are swimming, resort, beautiful. It is clear that the number of significant words can be modified changing threshold values.

2. *Ontology matching.* We match this words with specific sub-ontologies of Enterprise_Ontology and in particular with Cooking_Ontology, Restaurant_Ontology, General_Services_Ontology. The first ontology defines in detail all products in kitchen. The second all component and services in the restaurant and the last all terms relative to general services.

3. *Significant words disambiguation.* Statements with 3 or plus significant words similar in the specific ontology will be well-matched (SWM). There is a problem for statements with one or two significant word; in this case the statement will be non-well-matched (SNWM). For example the words *fish, serve* may be present in together ontologies Cooking_Ontology and Restaurant_Ontology but *fish, serve, customer* surely are only in the Restaurant_Ontology. To solve the statement disambiguation with one or two significant words we can use the normalized product scalar measure according the minimum or nearest-neighbour method:

$$S_n = S_y \qquad k > s$$

where

$$k = \frac{S_n . S_y}{\|S_n\| . \|S_y\|}$$

$S_n \in SNWM$, $S_y \in SWM$, $S_n \cdot S_y$ is the scalar product of two vectors and s is a value of a threshold.

This measure of similarity find the statement of SNWM nearest to SWM statement for the disambiguation and classification.

The error in our algorithm depends on the numerical vector, returned from TFIDF module, from significant words, that represent a specific statement, and from threshold s.

In Figure 6 is shown the routing of negative statements. After the matching with the Enterprise Ontology, each statement, with affective vector (h-s-f-sd-a-d–p), is categorized according to type of service to which it belongs. Then the statements belonging to the same service are aggregated in three distinct classes. At this point, a message containing compliant, with affective vector and polarity, is

automatically routed to individuated CCs.
In the case of enterprise with a lot of CCs, we can improve the effectiveness of the routing by sending messages only to most competent center that answers to many complaints.

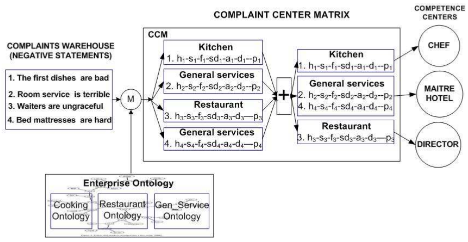

FIGURE 6. Negative statements routing

5. Actuation

The actuation phase is the less automatic one because the decision making process is mainly performed by people. In this phase CCs exploit the complaint messages for the improvement of products/services or for answering customers. In both cases, the enterprise give a (indirect or direct) feedback to the customer. As a matter of fact, the improvement of a product is an indirect message, communicating that the enterprise acknowledged the customer complaints. On the other hand, the enterprise may direct answer customer over the same collaborative channel used by the customer for expressing her opinions improving the effectiveness of communications towards the market. In this case, the enterprise becomes, in a peer-to-peer web 2.0 vision, an actor as a customer. A feedback line from CCs and CCM (dotted line in Fig. 1) allows managers to dynamically adjust the mapping of CCs to specific complaint.

6. Conclusion

In this work we propose the CeC model, a customer-centred enterprise information system model, aimed to exploit customer opinions to improve enterprise product/service. The CeC model is aimed to find, collect and analyse opinions, to react to stimuli of market and to send feedback to customers. The core of CeC is formed by the the Mapping phase and in particular from modules of Opinion Processing and Complaint Center Mapping. The Opinion Processing polarizes and filters opinions in complaint and Mapping module routes these complaints to competent centers to solve problem and satisfy the customer. These modules are very *mining-intensive* because implement various intelligent techniques of data mining, text mining, opinion mining and sentiment analysis. In our ongoing project, at the moment we are developing algorithm to map complaints to competence center that interfaces with our software of sentiment analysis, already tested, for polarizing customer opinions.

References

[1] J. I. Hagel and J. F. Rayport, *Creating value in the network economy*, ser. The Harvard Business Review Book Series. Boston, MA, USA: Harvard Business School Press, 1999, ch. The coming battle for customer information, pp. 159–171.

[2] Y. Cho, I.Im, R. Hiltz, and J. Fjermestad, "An analysis of online customer complaints: implications for web complaint management," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE, 7-10 Jan. 2002, pp. 2308–2317.

[3] S. Bolasco, A. Canzonetti, F. M. Capo, F. della Ratta-Rinaldi, and B. K. Singh, "Understanding text mining: A pragmatic approach," in *Knowledge Mining*, ser. Studies in Fuzziness and Soft Computing, S. Sirmakessis, Ed. Springer Verlag, 2005, vol. 185, pp. 31–50.

[4] D. Consoli, C. Diamantini, and D. Potena, "Improving the customer intelligence with customer enterprise customer model," in *Proceedings of the 10th International Conference on Enterprise Information Systems*, Barcelona, Spain, 2008, pp. 323–326.

[5] S. Kuhlins and A. Korthaus, "A multithreaded java framework for information extraction in the context of enterprise application integration," in *ISICT '03: Proceedings of the 1st international symposium on Information and communication technologies*. Trinity College Dublin, 2003, pp. 518–523.

[6] P. Ekman, *Emotions Revealed: Recognizing Faces and Feelings to Improve Communication and Emotional Life*. NY: OWL Books, 2007.

[7] C.-G. Ortony, A. and A. Collins, *The cognitive structure of emotions*. New York: Cambridge University Press, 1990.

[8] A. Esuli and F. Sebastiani, "SentiWordNet: A publicly available lexical resource for opinion mining," in *Proceedings of LREC-06, 5th Conference on Language Resources and Evaluation*, Genova, IT, 2006, pp. 417–422.

[9] C. Strapparava and A. Valitutti, "Wordnet-affect: An affective extension of wordnet," in *The 4th International Conference On Language Resources and Evaluation*, 2004, pp. 1083–1086.

[10] D. Consoli, C. Diamantini, and D. Potena, "A conceptual framework for web opinion mining," in *Proceedings 5th Conference of the Italian Chapter of AIS*, Paris, France, 2008b, pp. 1–8.

[11] M. Ikonomakis, S. Kotsiantis, and V. Tampakas, "Text classification: a recent overview," in *ICCOMP'05: Proceedings of the 9th WSEAS International Conference on Computers*. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2005, pp. 1–6.

[12] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, 2002.

[13] S. Qu, S. Wang, and Y. Zou, "Improvement of text feature selection method based on tfidf," in *FITME '08: Proceedings of the 2008 International Seminar on Future Information Technology and Management Engineering*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 79–81.

Dipartimento di Ingegneria
Informatica Gestionale e dell'Automazione
Universit Politecnica delle Marche
Ancona, Italy
Email:{consoli, diamantini, potena}@diiga.univpm.it