The final published version is available online at: https://link.springer.com/article/10.1007/s12559-014-9308-6

Propagating and Aggregating Fuzzy Polarities for Concept-Level Sentiment Analysis

Mauro Dragoni[1], Andrea G.B. Tettamanzi[2], and Célia da Costa Pereira[2]

[1] FBK–IRST, Trento, Italy
[2] Universite´ Nice Sophia Antipolis, I3S, UMR 7271, Sophia Antipolis, France
dragoni@fbk.eu andrea.tettamanzi|celia.pereira@unice.fr

Abstract. An emerging field within sentiment analysis concerns the investigation about how sentiment polarities associated with concepts have to be adapted with respect to the different domains in which they are used. In this paper, we explore the use of fuzzy logic for modeling concept polarities, and the uncertainty associated with them, with respect to different domains. The approach is based on the use of a knowledge graph built by combining two linguistic resources, namely WordNet and SenticNet. Such a knowledge graph is then exploited by a graph-propagation algorithm that propagates sentiment information learned from labeled datasets. The system implementing the proposed approach has been evaluated on the Blitzer dataset. The results demonstrate its viability in real-world cases.

1   Introduction

Sentiment analysis is a natural language processing task whose aim is to classify documents according to the opinion (polarity) they express on a given subject [1]. Generally speaking, sentiment analysis aims at determining the attitude of a speaker or a writer with respect to a topic or the overall tonality of a document. This task has created a considerable interest due to its wide applications. In recent years, the exponential increase of the Web for exchanging public opinions about events, facts, products, etc., has led to an extensive usage of sentiment analysis approaches, especially for marketing purposes.

By formalizing the sentiment analysis problem, a "sentiment" or "opinion" has been defined by [2] as a quintuple:

$$\langle o_j, f_{jk}, so_{ijkl}, h_i, t_l \rangle, \tag{1}$$

where o_j is a target object, f_{jk} is a feature of the object o_j, so_{ijkl} is the sentiment value of the opinion of the opinion holder h_i on feature f_{jk} of object o_j at time t_l, so_{ijkl} is positive (by denoting a state of happiness, bliss, or satisfaction), negative (by denoting a state of sorrow, dejection, or disappointment), or neutral (it is not possible to denote any particular sentiment), or a more granular rating, h_i is an opinion holder, t_l is the time when the opinion is expressed.

Such an analysis, may be "document-based", where the positive, negative, or neutral sentiment is assigned to the entire document content; or it may be "sentence-based"

where individual sentences are analyzed separately and classified according to the different polarity values. In the latter case, it is necessary to find with a high precision which are the entities towards which sentiments are directed in order to identify which are the adjectives (or, more in general, concepts) characterizing such entities.

In the classic sentiment analysis problem, the polarity of each term of the document is computed independently of the domain which the document belongs to. Recently, the idea of adapting terms polarity to different domains emerged [3]. The rationale behind the idea of such investigation is simple. Let us consider the following example concerning the adjective "small":

1. The sideboard is **small** and it is not able to contain a lot of stuff.
2. The **small** dimensions of this decoder allow to move it easily.

In the first sentence, we considered the Furnishings domain and, within it, the polarity of the adjective "small" is, for sure, "negative" because it highlights an issue of the described item. On the other hand, in the second sentence, where we considered the Electronics domain, the polarity of such an adjective may be considered "positive".

Unlike the approaches already discussed in the literature (and presented in Section 2), we address the multi-domain sentiment analysis problem by applying fuzzy set theory. We model the relationships between concepts and domains as fuzzy relations. Moreover, the proposed system exploits the use of semantic background knowledge for propagating information represented by the learned fuzzy membership functions to each element of the network. To the best of our knowledge, the proposed approach is innovative with respect to the state of the art of multi-domain sentiment analysis.

The rest of the article is structured as follows. Section 2 presents a survey on works about sentiment analysis. Section 3 provides a background on fuzzy set theory by presenting the elements used in our work. Section 4 introduces the background knowledge and tools used during the development of the system that is described in detail in Section 5. Section 6 reports the system evaluation performed on the Blitzer dataset and, finally, Section 7 concludes the article.

2 Related Work

The topic of sentiment analysis has been studied extensively in the literature [4,2], where several techniques have been proposed and validated.

Machine learning techniques are the most common approaches used for addressing this problem, given that any existing supervised methods can be applied to sentiment classification. For instance, in [1] and [5], the authors compared the performance of Naive-Bayes, Maximum Entropy, and Support Vector Machines in sentiment analysis on different features like considering only unigrams, bigrams, combination of both, incorporating parts of speech and position information or by taking only adjectives. Moreover, beside the use of standard machine learning method, researchers have also proposed several custom techniques specifically for sentiment classification, like the use of adapted score function based on the evaluation of positive or negative words in product reviews [6], as well as by defining weighting schemas for enhancing classification accuracy [7].

An obstacle to research in this direction is the need of labeled training data, whose preparation is a time-consuming activity. Therefore, in order to reduce the labeling effort, opinion words have been used for training procedures. In [8] and [9], the authors used opinion words to label portions of informative examples for training the classifiers. Opinion words have been exploited also for improving the accuracy of sentiment classification, as presented in [10], where a framework incorporating lexical knowledge in supervised learning to enhance accuracy has been proposed. Opinion words have been used also for unsupervised learning approaches like the ones presented in [11] and [12].

Another research direction concerns the exploitation of discourse-analysis techniques. [13] and [14] discuss some discourse-based supervised and unsupervised approaches for opinion analysis; while in [15], the authors present an approach to identify discourse relations.

The approaches presented above are applied at the document-level, i.e., the polarity value is assigned to the entire document content. However, for improving the accuracy of the sentiment classification, a more fine-grained analysis of the text, i.e., the sentiment classification of the single sentences, has to be performed. In the case of sentence-level sentiment classification, two different sub-tasks have to be addressed: (i) to determine if the sentence is subjective or objective, and (ii) in the case that the sentence is subjective, to determine if the opinion expressed in the sentence is positive, negative, or neutral. The task of classifying a sentence as subjective or objective, called "subjectivity classification", has been widely discussed in the literature [16,17,18,19,20]. Once subjective sentences are identified, the same methods as for sentiment classification may be applied. For example, in [21] the authors consider gradable adjectives for sentiment spotting; while in [22] and [23] the authors built models to identify some specific types of opinions.

The growth of product reviews was the perfect floor for using sentiment analysis techniques in marketing activities. However, the issue of improving the ability of detecting the different opinions concerning the same product expressed in the same review became a challenging problem. Such a task has been faced by introducing "aspect" extraction approaches that were able to extract, from each sentence, which is the aspect the opinion refers to. In the literature, many approaches have been proposed: conditional random fields (CRF) [24,25], hidden Markov models (HMM) [26,27,28], sequential rule mining [29], dependency tree kernels [30], and clustering [31]. In [32,33], a method was proposed to extract both opinion words and aspects simultaneously by exploiting some syntactic relations of opinion words and aspects.

A particular attention should be given also to the application of sentiment analysis in social networks. More and more often, people use social networks for expressing their moods concerning their last purchase or, in general, about new products. Such a social network environment opened up new challenges due to the different ways people express their opinions, as described by [34] and [35], who mention "noisy data" as one of the biggest hurdles in analyzing social network texts.

One of the first studies on sentiment analysis on micro-blogging websites has been discussed in [36], where the authors present a distant supervision-based approach for sentiment classification.

At the same time, the social dimension of the Web opens up the opportunity to combine computer science and social sciences to better recognize, interpret, and process opinions and sentiments expressed over it. Such multi-disciplinary approach has been called *sentic computing* [37]. Application domains where sentic computing has already shown its potential are the cognitive-inspired classification of images [38], of texts in natural language, and of handwritten text [39].

Finally, an interesting recent research direction is domain adaptation, as it has been shown that sentiment classification is highly sensitive to the domain from which the training data is extracted. A classifier trained using opinionated documents from one domain often performs poorly when it is applied or tested on opinionated documents from another domain, as we demonstrated through the example presented in Section 1. The reason is that words and even language constructs used in different domains for expressing opinions can be quite different. To make matters worse, the same word in one domain may have positive connotations, but in another domain may have negative connotations; therefore, domain adaptation is needed. In the literature, different approaches related to the Multi-Domain sentiment analysis have been proposed. Briefly, two main categories may be identified: (i) the transfer of learned classifiers across different domains [40,41,3,42,43,44,45], and (ii) the use of propagation of labels through graph structures [46,47,48,49]. Independently of the kind of approach, works using concepts rather than terms for representing different sentiments have been proposed.

3 Background on Fuzzy Set Theory

In this section we provide basic definitions and results about fuzzy sets, which will be used in the rest of the article.

3.1 Basic Considerations

Fuzzy Sets Fuzzy sets [50], allow the representation of imprecise information. Information is imprecise when the value of the variable to which it refers cannot be completely determined within a given universe of discourse. Fuzzy sets are then a generalization of classical sets obtained by replacing the characteristic function of a set A, χ_A, which takes up values in $\{0, 1\}$ ($\chi_A(x) = 1$ iff $x \in A$, $\chi_A(x) = 0$ otherwise) with a *membership function* μ_A, which can take up any value in $[0, 1]$. The value $\mu_A(x)$ or, more simply, $A(x)$ is the membership degree of element x in A, i.e., the degree to which x belongs in A.

A fuzzy set is completely defined by its membership function. Therefore, it is useful to define a few terms describing various features of this function, summarized in Figure 1. Given a fuzzy set A, its *core* is the (conventional) set of all elements x such that $A(x) = 1$; its *support*, $\text{supp}(A)$, is the set of all x such that $A(x) > 0$. A fuzzy set is *normal* if its core is nonempty. The set of all elements x of A such that $A(x) \geq \alpha$, for a given $\alpha \in (0, 1]$, is called the α-cut of A, denoted A_α.

The usual set-theoretic operations of union, intersection, and complement can be defined as a generalization of their counterparts on classical sets by introducing two families of operators, called triangular norms and triangular co-norms. In practice, it is
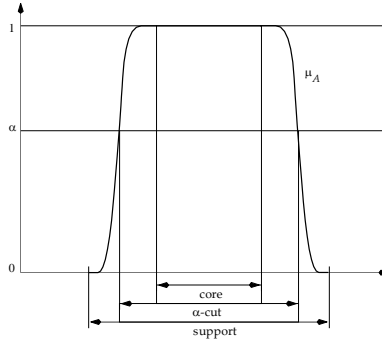
Fig. 1: Core, support, and α-cuts of a set A of the real line, having membership function μ_A.

usual to employ the min norm for intersection and the max co-norm for union. Given two fuzzy sets A and B, and an element x,

$$(A \cup B)(x) = \max\{A(x), B(x)\}; \tag{2}$$

$$(A \cap B)(x) = \min\{A(x), B(x)\}; \tag{3}$$

$$\bar{A}(x) = 1 - A(x). \tag{4}$$

Possibility Theory The membership function of a fuzzy set describes the more or less possible and mutually exclusive values of one (or more) variable(s). Such a function can then be seen as a possibility distribution [51]. Indeed, if F designates the fuzzy set of possible values of a variable X, $\pi_X = \mu_F$ is called the possibility distribution associated to X. The identity $\mu_F(v) = \pi_X(v)$ means that the membership degree of v to F is equal to the possibility degree of X being equal to v when all we know about X is that its value is in F. A possibility distribution for which there exists a completely possible value ($\exists v_0; \pi(v_0) = 1$) is said to be *normalized*.

3.2 The Extension Principle

The extension principle [52] is the main formal tool for making any mathematical theory fuzzy in a consistent and well-founded way.

Let U be the Cartesian product of n universes $U_1, \ldots, U_n$ and let $A_1, \ldots, A_n$ be an equal number of fuzzy sets defined in $U_1, \ldots, U_n$ respectively.

Suppose $t : U \longrightarrow V$ is a morphism from U into a new universe V. The question we ask is what the image of a fuzzy subset of U in this new universe V would be under the morphism t. This image would also be a fuzzy set, and its membership function would be calculated from the membership function of the original set and the morphism t.

Let B represent the fuzzy set induced in V by morphism t from the fuzzy sets $A_1, \ldots, A_n$ defined in U. The Extension Principle states that B has membership function, for all $y \in V$,

$$\mu_B(y) = \sup_{(x_1, \ldots, x_n) \in t^{-1}(y)} \min\{\mu_{A_1}(x_1), \ldots, \mu_{A_n}(x_n)\}. \tag{5}$$

B is said to extend fuzzy sets $A_1, \ldots, A_n$ in V.

Equation 5 is expressed for morphisms t of general form. If t is a discrete-valued function, the sup operator can be replaced by the max operator.

3.3 Defuzzification Methods

There may be situations in which the output of a fuzzy inference needs to be a crisp number $y*$ instead of a fuzzy set R. Defuzzification is the conversion of a fuzzy quantity into a precise quantity.

At least seven methods in the literature are popular for defuzzifying fuzzy outputs [53], which are appropriate for different application contexts. The *centroid method* is the most prominent and physically appealing of all the defuzzification methods. It results in a crisp value

$$y* = \frac{\int y\mu_R(y)dy}{\int \mu_R(y)dy},$$
(6)

where the integration can be replaced by summation in discrete cases.

4 Material

The proposed approach exploits a background knowledge for representing the linguistic information concerning each "concept" and the relationships between them. Such a background knowledge has been built incrementally by aggregating different resources freely available to the research community. Below, we list the resources used and decribe how they have been integrated for composing the final knowledge based used in the system.

4.1 WordNet

WordNet[3] [54] is a large lexical database of English nouns, verbs, adjectives, and adverbs grouped into sets of cognitive synonyms called *synsets*, where each synset expresses a distinct concept. In particular, each synset represents a list of synonyms, intended as words that denote the same concept and that are interchangeable in many contexts. WordNet contains around 117,000 synsets linked to each other by a small set of "conceptual relations", i.e., synonymy, hypernymy, hyponymy, etc..

Additionally, a synset contains a brief definition ("gloss") and, in most cases, one or more short sentences illustrating the use of the synset members. Words having several distinct meanings are represented in as many distinct synsets.

Even if WordNet superficially resembles a thesaurus, there are some important distinctions with respect to it. Firstly, WordNet does not define links between words, but between specific senses of words; this way, words that are found in close proximity to one another in the network are semantically disambiguated. Secondly, WordNet labels the semantic relations among words, whereas the groupings of words in a thesaurus does not follow any explicit pattern other than the similarity of their meanings.

In the implemented system, WordNet has been used as a starting point for the construction of the semantic graph which is used by our method (see Section 5).

---

[3] https://wordnet.princeton.edu/

4.2 SenticNet

SenticNet[4] [55] is a publicly available resource for opinion mining that exploits both artificial intelligence and semantic Web techniques to infer the polarities associated with common-sense concepts and to represent them in a semantic-aware format. In particular, SenticNet uses dimensionality reduction to calculate the affective valence of a set of Open Mind[5] concepts and it represents them in a machine accessible and processable format.

The development of SenticNet was inspired by SentiWordNet [56], a lexical resource in which each WordNet synset is associated to three numerical scores describing how objective, positive, and negative the terms contained in each synset are. The differences between SenticNet and SentiWordNet are basically three: (i) in SentiWordNet, each synset is associated to a three-valued representation (the objectivity of the synset, its positiveness, and its negativeness), while in SenticNet there is only one value belonging to the $[-1, 1]$ interval for representing the polarity of the concept; (ii) SenticNet provides the sentiment model of more complex common-sense concepts, while SentiWordNet is focused on assigning polarities to WordNet synsets: for instance, in SenticNet, complex concepts like "make good impression", "look attractive", "show appreciation" , "being fired", "leave behind", or "lose control" are used for defining positive or negative situations; and (iii) completely neutral concepts are not reported.

In order to represent SenticNet in a machine-accessible and processable way, information about each concept is encoded as a set of RDF triples using an XML syntax. In particular, concepts are identified using the ConceptNet Web API and statements, which have the form "concept-hasPolarity-polarityValue", are encoded in the RDF/XML format on the basis of the human emotion ontology (HEO) [57], a high-level ontology of human emotions, which supplies the most significant concepts and properties which constitute the centerpiece for the description of every human emotion.

As an example, the representation of the concept "a lot of fun" contained in SenticNet is shown below:

```
<rdf:Description rdf:about="http://sentic.net/api/en/concept/a_lot_of_fun">
  <rdf:type rdf:resource="http://sentic.net/api/concept"/>
  <text xmlns="http://sentic.net/api">a lot of fun</text>
  <semantics xmlns="http://sentic.net/api"
    rdf:resource="http://sentic.net/api/en/concept/radical"/>
  <semantics xmlns="http://sentic.net/api"
    rdf:resource="http://sentic.net/api/en/concept/enjoyable"/>
  <semantics xmlns="http://sentic.net/api"
    rdf:resource="http://sentic.net/api/en/concept/just_fun"/>
  <semantics xmlns="http://sentic.net/api"
    rdf:resource="http://sentic.net/api/en/concept/good_mental_health"/>
  <semantics xmlns="http://sentic.net/api"
    rdf:resource="http://sentic.net/api/en/concept/fun_play"/>
  <pleasantness xmlns="http://sentic.net/api"
    rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.814</pleasantness>
  <attention xmlns="http://sentic.net/api"
    rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0</attention>
  <sensitivity xmlns="http://sentic.net/api"
    rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0</sensitivity>
  <aptitude xmlns="http://sentic.net/api"
    rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.856</aptitude>
```

---

[4] http://sentic.net/

[5] http://commons.media.mit.edu/en/

```
  <polarity xmlns="http://sentic.net/api"
     rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.557</polarity>
</rdf:Description>
```

SenticNet contains more than 5,700 polarity concepts (nearly 40% of the Open Mind corpus) and it may be connected with any kind of opinion mining application. For example, after the de-construction of the text into concepts through a semantic parser, SenticNet can be used to associate polarity values to these and, hence, infer the overall polarity of a clause, sentence, paragraph, or document by averaging such values.

4.3 Resource Integration

The two resources discussed above have been linked in order to build a single knowledge graph used for propagating information through concepts. The creation of the links between the two resources has been done by taking into account both advantages and drawback of making such a connection: the main advantage is that the integration of SenticNet allows to increase the coverage of terms due to the lack of common-sense concepts in WordNet; the main drawback is that the creation of such links may lead to the injection of ambiguities into the knowledge graph. Therefore, particular attention has been paid to when links between SenticNet concepts and WordNet synsets were to be defined.

As a consequence, the creation of the links between the two linguistic resources has been subjected to the following constraints:

– if the WordNet term has only one synset, it means that the term is not ambiguous; therefore, the link may be created;
– if the WordNet term is associated to more than one synset, the synonyms of the SenticNet concept are intersected with the synonyms of each term's synset: if only one synset has synonyms in common with the SenticNet concept's synonyms, the link is created; otherwise it is not.

5   Method

The main goal of the implemented system is to learn fuzzy membership functions representing the degree to which a concept belongs to a domain in terms of both sentiment polarity and aboutness. The two pillars on which the system has been conceived are: (i) the use of fuzzy logic for modeling the polarity of a concept with respect to a domain as well as its aboutness, and (ii) the creation of a two-level graph where the top level represents the semantic relationships between concepts, while the bottom level contains the links between all concept membership functions and the domains.

Figure 2 shows the conceptualization of such two-level graph. Relationships between the concepts of the Level 1 (the Semantic Level) are described by the background knowledge exploited by the system as described in Section 4. The type of relationships are the same generally used in linguistic resource: for example, concepts C_1 and C_3 may be connected through an is-a relationship as well as an antonym relationship. Instead, each connection of the Level 2 (the Sentiment Level) describes the degree of membership of each concept to the different domains taken into account.

The system has been trained using the Blitzer dataset[6] in two steps: first, the fuzzy membership functions have been initially estimated by analyzing only the explicit information present within the dataset (Section 5.1); then, (ii) the explicit information has been propagated through the Sentiment Level graph by exploiting the connections defined in the Semantic Level.
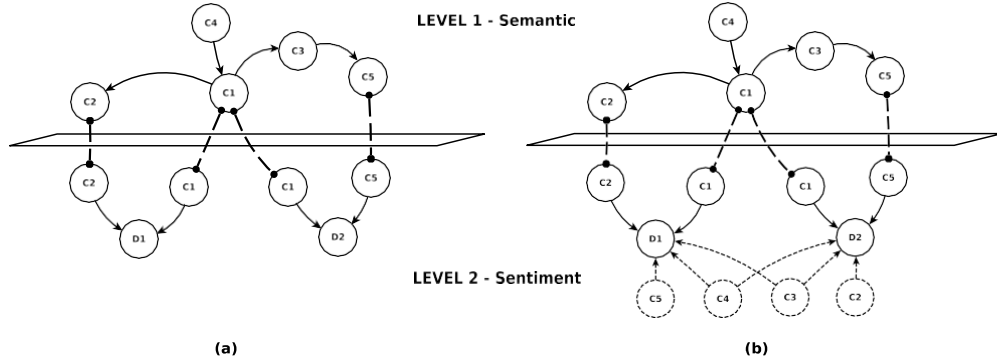

Fig. 2: The two-layer graph initialized during the Preliminary Learning Phase (a) and its evolution after the execution of the Information Propagation Phase (b).

5.1 Preliminary Learning Phase

The Preliminary Learning (PL) phase aims at estimating the starting polarity of each concept with respect to a domain. The estimation of this value is done by analyzing only the explicit information provided by the training set. This phase allows to define the preliminary fuzzy membership functions between the concepts defined in the Semantic Level of the graph and the domains that are defined in the Sentiment one. Such a value is computed as

$$\text{polarity}_i^{(0)}(C) = \frac{k_C^i}{T_C^i} \in [-1, 1] \qquad \forall i = 1, \ldots, n, \tag{7}$$

where C is the concept taken into account, index i refers to domain D_i which the concept belongs to, n is the number of domains available in the training set, k_C^i is the arithmetic sum of the polarities observed for concept C in the training set restricted to domain D_i, and T_C^i is the number of instances of the training set, restricted to domain D_i, in which concept C occurs. The shape of the fuzzy membership function generated during this phase is a triangle with the top vertex in the coordinates $(x, 1)$, where $x = \text{polarity}_i^{(0)}(C)$ and with the two bottom vertices in the coordinates $(-1, 0)$ and $(1, 0)$ respectively. The rationale is that while we have one point (x) in which we have

_____

[6] http://www.cs.jhu.edu/ mdredze/datasets/sentiment/

full confidence, our uncertainty covers the entire space because we do not have any information concerning the remaining polarity values.

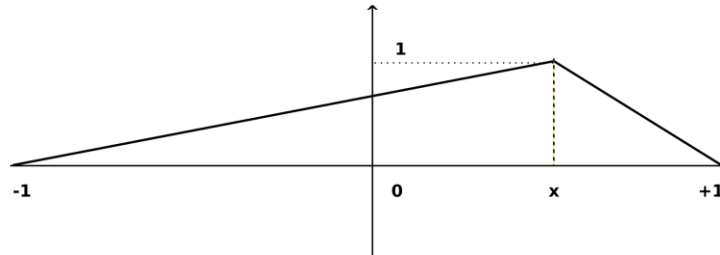Figure 3 shows a picture of the generated fuzzy triangle.

Fig. 3: The fuzzy triangle generated after the Preliminary Learning Phase.

5.2 Information Propagation Phase

The Information Propagation (IP) phase aims at exploiting the explicit information learned in the PL phase in order to both (i) refine the fuzzy membership function of the known concepts and (ii) model such functions for concepts that are not specified in the training set, but are semantically related to the specified ones. Figure 2 shows how the two-level graph evolves before and after the execution of the IP phase. After the PL phase, only four membership functions are modeled: C_1 and C_2 for domain D_1, and C_1 and C_5 for domain D_2 (Figure 2a). However, as we may observe, in the Semantic Level there are concepts that are semantically related to the ones that were explicitly defined in the training set, namely C_3 and C_4; furthermore, there are concepts for which a fuzzy membership function has not been modeled for some domains (i.e. C_2 for domain D_2 and C_5 for domain D_1).

Such fuzzy membership functions may be inferred by propagating the information modeled in the PL phase. Similarly, existing fuzzy membership functions are refined by the influence of the other ones. Let us consider the polarity of concept C_3 in domain D_2. The fuzzy membership function representing this polarity is strongly influenced by the ones representing the polarities of concepts C_1 and C_5 with respect to domain D_2. Furthermore, both the polarities of C_1 and C_5 will be refined during the IP phase.

The propagation of the learned information through the graph is done iteratively. At each iteration $t = 1, 2, \ldots$, the initial estimated polarity value $\text{polarity}_i^{(0)}(C)$ of concept C for domain D_i, learned during the PL phase, is updated based on the learned values of the adjoining concepts $C_1, \ldots, C_n$. This process is carried out for all domains D_i in parallel and independently. Therefore, in the following explanation, we will omit the subscript i whenever it is not necessary, keeping in mind that polarity propagation for one domain does not interact with polarity propagation for the other domains.

At iteration $t + 1$, the estimated polarity of concept C is updated as follows:

$$\text{polarity}^{(t+1)}(C) = (1-\lambda)\text{polarity}^{(t)}(C) + \lambda \frac{1}{n}\sum_{j=1}^{n}\text{polarity}^{(t)}(C)_j \qquad (8)$$

where $C_1, \ldots, C_n$ are the concepts adjacent to C, and $0 < \lambda < 1$ is a parameter of the algorithm, which we shall call *propagation rate*.

This iterative process stops as soon as the sum of the variations of the polarity for each concept and domain,

$$\Delta_t = \sum_C \sum_i \text{polarity}^{(t)}(C) - \text{polarity}^{(t-1)}_i(C) \qquad (9)$$

falls below a threshold L, which is another parameter of the algorithm we will call *convergence limit*.

At each iteration $t = 0, 1, 2, \ldots$, the values $\text{polarity}^{(t)}(C)$ are saved in order to exploit them for the calculation of the shapes of the fuzzy membership functions associating concept C to domain D_i.

The final shapes of the inferred fuzzy membership functions at the end of the IP phase are trapezoids whose core consists of the interval between the polarity value learned during the PL phase, $\text{polarity}^{(0)}_i(C)$, and the final polarity value at the end of the IP phase, $\text{polarity}^{(t_{\text{stop}})}_i(C)$ and whose support extends beyond the core on either side by half the variance $\sigma^2_{C,i}$ of the distribution of $\text{polarity}^{(t)}(C)$, $t = 0, \ldots, t_{\text{stop}}$. To sum up, for each domain D_i, $\mu_{C,i}$ is a trapezoid with parameters (a, b, c, d), where

$$a = \min\{\text{polarity}^{(0)}_i(C), \text{polarity}^{(t_{\text{stop}})}_i(C)\},$$
$$b = \max\{\text{polarity}^{(0)}_i(C), \text{polarity}^{(t_{\text{stop}})}_i(C)\},$$
$$c = \max\{-1, a - \sigma^2_{C,i}/2\},$$
$$d = \min\{1, b + \sigma^2_{C,i}/2\}.$$

The idea here is that the most likely values for the polarity of C for domain D_i are those comprised between the initial and final value of the IP phase and the more quickly the polarity values converged during the IP phase, the least uncertainty there is about the resulting polarity estimate. Conversely, a polarity value that converged slowly or with many fluctuations is going to yield a less reliable, and thus more uncertain, estimate.

Figure 4 shows a picture of the generated fuzzy trapezoid.

5.3 Polarity Aggregation Phase

The fuzzy polarities of different concepts, resulting from the IP phase, are finally aggregated by a fuzzy averaging operator obtained by applying the extension principle (cf. Section 3.2) in order to compute fuzzy polarities for complex entities, like texts, which consist of a number of concepts and thus derive, so to speak, their polarity from them. When a crisp polarity value is needed, it may be computed from a fuzzy polarity by applying one of the defuzzification methods described in Section 3.3.
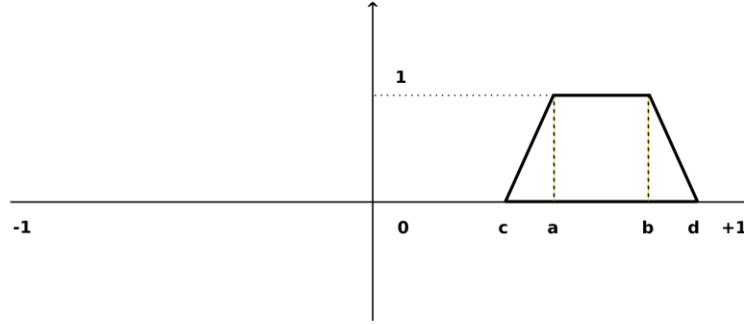
Fig. 4: The fuzzy trapezoid generated after the Information Learning Phase.

Let $\mu_C : [-1, 1] \to [0, 1]$ be the fuzzy interval (i.e., a convex fuzzy set) representing the fuzzy polarity of concept C resulting from the IP phase. Let T be a text (or any other entity that may be regarded as a combination of concepts) related to concepts $C_1, \ldots, C_n$. The fuzzy polarity of T, $\mu_T : [-1, 1] \to [0, 1]$, may be defined as the average of the fuzzy polarities of concepts $C_1, \ldots, C_n$, by applying the extension principle, as follows, for all $x \in [-1, 1]$:

$$\mu_T(x) = \sup_{x = \frac{1}{n}\sum_{i=1}^{n} x_i} \min_{i=1,\ldots,n} \mu_{C_i}(x_i). \tag{10}$$

Computing $\mu_T(x)$ directly by Equation 10 would be impractical, as it would involve discretizing the values of each variable $x_i \in [-1, 1]$ with a step $\Delta x = \frac{1}{N}$, computing the average $x = \frac{1}{n}\sum_{i=1}^{n} x_i$ for each of the N^n possible n-tuples $\langle x_1, \ldots, x_n \rangle$ and then keeping track of the maximum value of $\min_{i=1,\ldots,n} \mu_{C_i}(x_i)$ for each x. Fortunately, a much more efficient method, if all the μ_{C_i} are convex, is to sample the ordinates, i.e., the degrees of membership. This method is known in the literature on fuzzy numbers as the DSW algorithm [58]. For every degree of membership $\alpha \in [0, 1]$, consider the α-cuts of $\mu_{C_1}, \ldots, \mu_{C_n}$, which are crisp intervals of the form $[a_i^\alpha, b_i^\alpha]$. The corresponding α-cut of μ_T is then given by

$$a_T^\alpha = \frac{1}{n}\sum_{i=1}^{n} a_i^\alpha, \quad b_T^\alpha = \frac{1}{n}\sum_{i=1}^{n} b_i^\alpha. \tag{11}$$

Notice that if N is the number of samples, only $2N$ averages will have to be computed here. Furthermore, if the μ_{C_i} are trapezoidal, we only need to compute the support ($\alpha = 0$) and the core ($\alpha = 1$) of μ_T, because all the intermediate α-cuts are linear combinations of them and the average is a linear operator. Therefore, if the μ_{C_i} are defined as trapezoids with parameters (a_i, b_i, c_i, d_i), then μ_T is also a trapezoid, with parameters

$$a_T = \frac{1}{n}\sum_{i=1}^{n} a_i, \quad b_T = \frac{1}{n}\sum_{i=1}^{n} b_i, \quad c_T = \frac{1}{n}\sum_{i=1}^{n} c_i, \quad d_T = \frac{1}{n}\sum_{i=1}^{n} d_i. \tag{12}$$

5.4 Decision

The result of the polarity aggregation phase is a fuzzy polarity, whose membership function reflects the uncertainty of the available estimate obtained by the system. In this sense, μ_T may be regarded as a possibility distribution of the actual polarity of T. Given $x \in [-1, 1]$, the membership degree $\mu_T(x)$ represent the degree to which it is possible that the polarity of T is x. Here, we are making the assumption that polarity is gradual, i.e., that a text may be more or less negative or positive.

At some point, if a decision must be made based on the polarity of T, some criterion has to be adopted, which takes the uncertainty of the estimate into account. The fact is a criterion can be defined only with reference to a given application scenario. For instance, if we can afford any desired number of texts and what we want is to pick a few of them whose polarity is certain, we can look for T such that either $d_T < 0$ or $a_T > 0$, i.e., the support of μ_T lies entirely on the left or on the right of zero, because in those cases it is certain that polarity is negative (in the former case) or positive (in the latter). In other scenarios, where what we want is to classify each and every text as either negative or positive as accurately as possible, we will have to be less picky and rely on a defuzzification method to transform μ_T into a crisp polarity value.

6 System Evaluation

The proposed system has been evaluated on the Blitzer dataset [7] [3], a multi-domain sentiment analysis dataset containing reviews of Amazon products belonging to 25 different categories. The dataset is provided in two variants: (i) a balanced dataset, where for each category the number of negative and positive reviews is the same; and (ii) an unbalanced version where the proportion between positive and negative reviews has not been maintained. We decided to evaluate the system by using the balanced version of the dataset in order not to bias the system during the training phase and to avoid overfitting during the validation of the system. For the initialization and the training of the system we used 75% of the data; the remaining 25% has been used for validating it.

Concerning the training of the system, in the previous section we already introduced the "propagation rate" λ as the factor that leads the quantity of information exchanged by two concepts during the propagation phase of the algorithm and the "convergence limit" L, used for stopping the the polarity propagation through the knowledge graph. Such parameters are not the only ones whose value affects the effectiveness of the approach. Indeed, there is a further parameter that is used for managing the propagation phase of the algorithm, namely the "annealing rate". This parameter is used for decreasing, at each iteration of the algorithm, the value of the propagation rate. The rationale behind this parameter is the same used in the simulated annealing algorithm [59]; this way, at each iteration, the polarity propagation coefficient between concepts is reduced. Low values of the annealing rate, in general, shorten the execution of the propagation algorithm in terms of computational time.

Different combinations of these three parameter values have been used in order to observe how the system if affected by them. Before validating the system on the full

---

[7] http://www.cs.jhu.edu/~mdredze/datasets/sentiment/

Blitzer dataset, we performed an *in-vitro* experiment on a small portion of the dataset for defining which values would be used for the final evaluation. Table 1 shows the whole set of values assigned to each parameter. In the first column, we present the set of values combined for measuring the effectiveness of the system in the *in-vitro* use case; in the second column we reported the parameter values adopted for the final evaluation of the system.

Parameter	In-Vitro values set	Validation values
Annealing Rate	0.0; 0.1; 0.2; 0.3; 0.4; 0.5; 0.6; 0.7; 0.8; 0.9; 1.0	0.5
Propagation Rate	0.2; 0.15; 0.1; 0.075; 0.05; 0.025; 0.01; 0.005; 0.002	0.01
Convergence Value	10.0; 20.0; 100.0; 200.0; 500.0; 1000.0	20

Table 1: Set of parameter values used for the measuring how the effectiveness of the system changes based on how their are combined, and the values used for the final evaluation.

By analyzing the behavior of the system based on the combination of the parameter values, we observed that a trade-off between the number of iterations during the propagation of information phase and how much the information is propagated for each iteration is required. Indeed, the more the information is propagated through the graph, the higher is the risk of having all concept defined in the knowledge graph converge to the same polarity value. For this reason, to avoid that two concepts having opposite polarities, and having a long path connecting them, contaminate each other, a limit to the propagation phase is needed. The values used for the final evaluation of system allows not to overtake such a limit, in order to effectively propagate all information through the knowledge graph by having at the same time a polarity value for concepts that are not mentioned in the training set, as well as a barrier avoiding disruptive contamination between opposite concepts.

Such final values (i.e., annealing rate = 0.5; propagation rate = 0.01; convergence limit = 20) have been used for the final evaluation performed on the entire validation set. Table 2 presents such results. The system has been compared to three different baselines representing the best-known machine learning techniques available today.

By observing the results, we may state that the proposed approach obtained an improvement in accuracy with respect to the baselines. Besides the numerical comparison, where it is possible to see that in 17 out of 25 domains the proposed approach outperforms the baselines, there are some aspects that we would like to point out in order to analyze in more detail the behavior of the system.

If we observe the domains in which the proposed system outperforms the baselines with respect to the other ones, we may notice that, except for the "Computer, Video Games" domain, the worst performances are obtained on domains where only a small number of instances is available in the domain dataset. This may be seen by considering the number of validation instances, which represent the 25% of the instances contained in the domain dataset. On the contrary, by observing the results obtained for

Domain	# of Instances	SVN [60]	Naive-Bayes [61]	Max-Entropy [61]	MDFSA Precision	MDFSA Recall
Apparel	400	0.7975	0.8328	0.8200	**0.9100**	1.0000
Automotive	147	0.8095	0.7959	**0.8639**	0.6327	1.0000
Baby	380	0.7737	0.8105	0.7842	**0.9079**	0.9974
Beauty	299	0.7826	0.7391	0.8060	**0.8796**	1.0000
Books	400	0.7525	0.8275	0.7600	**0.9175**	0.9975
Camera, Photo	400	0.8200	0.8600	0.8400	**0.9025**	0.9975
Cell Phones Service	205	0.7756	0.7707	0.8341	**0.8341**	1.0000
Computer, Video Games	292	0.9555	0.9692	**0.9760**	0.8014	1.0000
DVD	400	0.7850	0.7775	0.7800	**0.9175**	0.9975
Electronics	400	0.7650	0.8225	0.7875	**0.9125**	0.9975
Gourmet Food	242	**0.8719**	0.8512	0.8678	0.7066	1.0000
Grocery	270	0.8630	0.7852	0.8444	**0.9148**	1.0000
Health Personal Care	400	0.8000	0.8525	0.8450	**0.8925**	0.9975
Jewelry Watches	258	0.8488	0.7829	**0.8760**	0.7868	1.0000
Kitchen Housewares	400	0.8225	0.8525	0.8550	**0.9050**	0.9975
Magazines	394	0.8096	0.8122	0.8249	**0.8401**	0.9975
Music	400	0.7700	**0.8000**	0.7975	0.7975	0.9975
Musical Instruments	67	**0.8806**	0.8507	0.8507	0.7313	1.0000
Office Products	86	**0.8605**	0.8488	0.8488	0.8140	1.0000
Outdoor Living	265	0.8151	0.7660	0.8151	**0.8717**	1.0000
Software	383	0.8146	0.8381	0.8303	**0.8695**	1.0000
Sports Outdoors	400	0.7725	0.8325	0.8175	**0.8425**	1.0000
Tools Hardware	22	**1.0000**	**1.0000**	**1.0000**	0.4783	1.0000
Toys Games	400	0.7925	0.8300	0.8250	**0.9100**	0.9975
Video	400	0.7975	0.8125	0.8300	**0.8325**	0.9975
Weighted Average	-	0.8068	0.8227	0.8275	**0.8617**	0.9987

Table 2: Results obtained on the balanced version of the Blitzer dataset.

domains having the largest validation sets (i.e. 400 instances, which means 1200 instances for training), the proposed system always achieves the best performance. This result points to a possible weakness of the system: indeed, it appears that a minimum quantity of training instances is required in order to have an initial "good coverage" of the knowledge graph. Here, with the expression "good coverage", we mean a portion of the knowledge graph that is set with initial polarity values after the initialization phase. This way, it is possible to have a more effective propagation of polarity information through the entire graph during the propagation phase.

Another aspect is related to the recall values obtained by the system. Indeed, in some cases the system was not able to compute an overall polarity value for all validation instances (for some domains, it happens that one instance has been discarded from the evaluation procedure). After a more in-depth analysis of the behavior of the system on the validation set, we noticed that the missing predictions are mainly related to neutral polarity scenarios. In such cases, the system does not provide any result, with the consequence of decreasing the final recall value.

Finally, in a couple of cases, the system was not able to compute a final polarity of the text due to missing information in the knowledge graph. Indeed, it may happen that, after the propagation phase, some concepts do not have any information concerning their polarity for that particular domain. In this cases, the system is not able to infer anything about the polarity of the text, and it returns a neutral polarity value.

7   Conclusion And Future Work

In this article, we have presented an approach to multi-domain sentiment analysis based on the use of fuzzy logic for computing the polarity of each concept with respect to a particular domain. The approach uses a linguistic knowledge graph, created from the integration of WordNet and SenticNet, to which a propagation algorithm has been applied for exchanging information acquired from the training set to the entire set of concepts defined within the knowledge graph. For each concept of the created knowledge graphs, one for each domain in which the system is used, a fuzzy membership function is modeled for representing the polarity of the concept with respect to the domain for which the knowledge graph has been created. This way, it is possible to model uncertainty about concept polarities, which can be exploited during computation of the text sentiment.

The system has been validated on the full version of the Blitzer dataset, obtaining a higher accuracy than the baselines considered[8].

Further work on the proposed system will focus both on the enrichment of the knowledge graph as well as on the use of fuzzy membership functions. On the knowledge graph side, the use of new sources will be used for extending the current version of the graph; moreover, the use of a richer set of relationships will be investigated in order to improve how polarity information is propagated through the graph. On the fuzzy side, a more detailed comparison between the different ways of representing and aggregating concept polarities will be performed, in order to analyze how different techniques may affect the effectiveness of the system. Finally, we foresee the integration of a concept extraction approach in order to equip the system with the capability of extracting finer-grained information (i.e., single aspects and semantic information associated with them) which can be used during the training of the system.

References

1. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? sentiment classification using machine learning techniques. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Philadelphia, Association for Computational Linguistics (July 2002) 79–86
2. Liu, B., Zhang, L.: A survey of opinion mining and sentiment analysis. In Aggarwal, C.C., Zhai, C.X., eds.: Mining Text Data. Springer (2012) 415–463
3. Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In: ACL. (2007) 187–205

---

[8] Detailed results and tool demo are available at `http://dkmtools.fbk.eu/moki/demo/mdfsa/mdfsa_demo.html`.

4. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval **2**(1-2) (2008) 1–135
5. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: ACL. (2004) 271–278
6. Dave, K., Lawrence, S., Pennock, D.M.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: WWW. (2003) 519–528
7. Paltoglou, G., Thelwall, M.: A study of information retrieval weighting schemes for sentiment analysis. In: ACL. (2010) 1386–1395
8. Tan, S., Wang, Y., Cheng, X.: Combining learn-based and lexicon-based techniques for sentiment detection without using labeled examples. In: SIGIR. (2008) 743–744
9. Qiu, L., Zhang, W., Hu, C., Zhao, K.: Selc: a self-supervised model for sentiment classification. In: CIKM. (2009) 929–936
10. Melville, P., Gryc, W., Lawrence, R.D.: Sentiment analysis of blogs by combining lexical knowledge with text classification. In: KDD. (2009) 1275–1284
11. Taboada, M., Brooke, J., Tofiloski, M., Voll, K.D., Stede, M.: Lexicon-based methods for sentiment analysis. Computational Linguistics **37**(2) (2011) 267–307
12. Turney, P.D.: Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In: ACL. (2002) 417–424
13. Somasundaran, S.: Discourse-level relations for Opinion Analysis. PhD thesis, University of Pittsburgh (2010)
14. Asher, N., Benamara, F., Mathieu, Y.Y.: Distilling opinion in discourse: A preliminary study. In: COLING (Posters). (2008) 7–10
15. Wang, H., Zhou, G.: Topic-driven multi-document summarization. In: IALP. (2010) 195–198
16. Riloff, E., Patwardhan, S., Wiebe, J.: Feature subsumption for opinion analysis. In: EMNLP. (2006) 440–448
17. Wiebe, J., Wilson, T., Bruce, R.F., Bell, M., Martin, M.: Learning subjective language. Computational Linguistics **30**(3) (2004) 277–308
18. Wilson, T., Wiebe, J., Hwa, R.: Just how mad are you? finding strong and weak opinion clauses. In: AAAI. (2004) 761–769
19. Wilson, T., Wiebe, J., Hwa, R.: Recognizing strong and weak opinion clauses. Computational Intelligence **22**(2) (2006) 73–99
20. Yu, H., Hatzivassiloglou, V.: Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. EMNLP '03, Stroudsburg, PA, USA, Association for Computational Linguistics (2003) 129–136
21. Hatzivassiloglou, V., Wiebe, J.: Effects of adjective orientation and gradability on sentence subjectivity. In: COLING. (2000) 299–305
22. Kim, S.M., Hovy, E.H.: Crystal: Analyzing predictive opinions on the web. In: EMNLP-CoNLL. (2007) 1056–1064
23. Kim, S.M., Pantel, P., Chklovski, T., Pennacchiotti, M.: Automatically assessing review helpfulness. In: EMNLP. (2006) 423–430
24. Jakob, N., Gurevych, I.: Extracting opinion targets in a single and cross-domain setting with conditional random fields. In: EMNLP. (2010) 1035–1045
25. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML. (2001) 282–289
26. Freitag, D., McCallum, A.: Information extraction with hmm structures learned by stochastic optimization. In: AAAI/IAAI. (2000) 584–589
27. Jin, W., Ho, H.H.: A novel lexicalized HMM-based learning framework for web opinion mining. In: Proceedings of the 26th Annual International Conference on Machine Learning. ICML '09, New York, NY, USA, ACM (2009) 465–472

28. Jin, W., Ho, H.H., Srihari, R.K.: Opinionminer: a novel machine learning system for web opinion mining and extraction. In: KDD. (2009) 1195–1204
29. Liu, B., Hu, M., Cheng, J.: Opinion observer: analyzing and comparing opinions on the web. In: WWW. (2005) 342–351
30. Wu, Y., Zhang, Q., Huang, X., Wu, L.: Phrase dependency parsing for opinion mining. In: EMNLP. (2009) 1533–1541
31. Su, Q., Xu, X., Guo, H., Guo, Z., Wu, X., Zhang, X., Swen, B., Su, Z.: Hidden sentiment association in chinese web opinion mining. In: WWW. (2008) 959–968
32. Qiu, G., Liu, B., Bu, J., Chen, C.: Expanding domain sentiment lexicon through double propagation. In: IJCAI. (2009) 1199–1204
33. Qiu, G., Liu, B., Bu, J., Chen, C.: Opinion word expansion and target extraction through double propagation. Computational Linguistics **37**(1) (2011) 9–27
34. Barbosa, L., Feng, J.: Robust sentiment detection on twitter from biased and noisy data. In: COLING (Posters). (2010) 36–44
35. Bermingham, A., Smeaton, A.F.: Classifying sentiment in microblogs: is brevity an advantage? In: CIKM. (2010) 1833–1836
36. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Standford University (2009)
37. Cambria, E., Hussain, A.: Sentic Computing: Techniques, Tools, and Applications. Volume 2 of SpringerBriefs in Cognitive Computation. Springer, Dordrecht, Netherlands (2012)
38. Cambria, E., Hussain, A.: Sentic album: Content-, concept-, and context-based online personal photo management system. Cognitive Computation **4**(4) (2012) 477–496
39. Wang, Q.F., Cambria, E., Liu, C.L., Hussain, A.: Common sense knowledge for handwritten chinese recognition. Cognitive Computation **5**(2) (2013) 234–242
40. Aue, A., Gamon, M.: Customizing sentiment classifiers to new domains: a case study. In: Proceedings of RANLP. (2005)
41. Yang, H., Callan, J., Si, L.: Knowledge transfer and opinion detection in the TREC 2006 blog track. In: TREC. (2006)
42. Pan, S.J., Ni, X., Sun, J.T., Yang, Q., Chen, Z.: Cross-domain sentiment classification via spectral feature alignment. In: WWW. (2010) 751–760
43. Bollegala, D., Weir, D.J., Carroll, J.A.: Cross-domain sentiment classification using a sentiment sensitive thesaurus. IEEE Trans. Knowl. Data Eng. **25**(8) (2013) 1719–1731
44. Xia, R., Zong, C., Hu, X., Cambria, E.: Feature ensemble plus sample selection: Domain adaptation for sentiment classification. IEEE Int. Systems **28**(3) (2013) 10–18
45. Yoshida, Y., Hirao, T., Iwata, T., Nagata, M., Matsumoto, Y.: Transfer learning for multiple-domain sentiment analysis—identifying domain dependent/independent word polarity. In: AAAI. (2011) 1286–1291
46. Ponomareva, N., Thelwall, M.: Semi-supervised vs. cross-domain graphs for sentiment analysis. In: RANLP. (2013) 571–578
47. Tsai, A.C.R., Wu, C.E., Tsai, R.T.H., jen Hsu, J.Y.: Building a concept-level sentiment dictionary based on commonsense knowledge. IEEE Int. Systems **28**(2) (2013) 22–30
48. Tai, Y.J., Kao, H.Y.: Automatic domain-specific sentiment lexicon generation with label propagation. In: iiWAS, ACM (2013) 53:53–53:62
49. Huang, S., Niu, Z., Shi, C.: Automatic construction of domain-specific sentiment lexicon based on constrained label propagation. Knowl.-Based Syst. **56** (2014) 191–200
50. Zadeh, L.A.: Fuzzy sets. Information and Control **8** (1965) 338–353
51. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems **1** (1978) 3–28
52. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning - i. Inf. Sci. **8**(3) (1975) 199–249

53. Hellendoorn, H., Thomas, C.: Defuzzification in fuzzy controllers. Intelligent and Fuzzy Systems **1** (1993) 109–123
54. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA (1998)
55. Cambria, E., Olsher, D., Rajagopal, D.: Senticnet 3: A common and common-sense knowledge base for cognition-driven sentiment analysis. In: AAAI. (2014) 1515–1521
56. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: LREC. (2010) 2200–2204
57. Grassi, M.: Developing heo human emotions ontology. In: COST 2101/2102 Conference. (2009) 244–251
58. Dong, W., Shah, H., Wong, F.: Fuzzy computations in risk and decision analysis. Civ. Eng. Syst. **2** (1985) 201–208
59. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. SCIENCE **220**(4598) (1983) 671–680
60. Chang, C.C., Lin, C.J.: Libsvm: A library for support vector machines. ACM TIST **2**(3) (2011) 27:1–27:27
61. McCallum, A.K.: Mallet: A machine learning for language toolkit. `http://mallet.cs.umass.edu` (2002)