Noname manuscript No.
(will be inserted by the editor)

Good Location, Terrible Food: Detecting Feature Sentiment in User-Generated Reviews

Mario Cataldi · Andrea Ballatore · Ilaria Tiddi · Marie-Aude Aufaure

Received: date / Accepted: date

Abstract A growing corpus of online informal reviews is generated every day by non-experts, on social networks and blogs, about an unlimited range of products and services. Users do not only express holistic opinions, but often focus on specific features of their interest. The automatic understanding of "what people think" at the feature level can greatly support decision making, both for consumers and producers. In this paper, we present an approach to feature-level sentiment detection that integrates natural language processing with statistical techniques, in order to extract users' opinions about specific features of products and services from user-generated reviews. First, we extract domain features, and each review is modelled as a lexical dependency graph. Second, for each review, we estimate the polarity relative to the features by leveraging the syntactic dependencies between the terms. The approach is evaluated against a ground truth consisting of set of user-generated reviews, manually annotated by 39 human subjects and available online, showing its human-like ability to capture feature-level opinions.

M. Cataldi
École Centrale Paris
Paris, France
E-mail: mario.cataldi@ecp.fr

A. Ballatore
University College Dublin
Dublin, Ireland
E-mail: andrea.ballatore@ucd.ie

I. Tiddi
Knowledge Media Institute, The Open University
Milton Keynes, UK
E-mail: ilaria.tiddi@open.ac.uk

M-A. Aufaure
École Centrale Paris
Paris, France
E-mail: marie-aude.aufaure@ecp.fr

Keywords Sentiment Analysis · Opinion Mining · Natural Language Processing · Feature Detection · Dependency Graphs

1 Introduction

One of the key aspects that has led to the popular success of web 2.0 is the possibility for the users to express unmediated individual opinions. On social networks and specialized web sites, millions of users express their opinion on a wide range of products and services. According to a survey by marketing firm comScore, almost 81% of Internet users have done online research on a product at least once and, among them, up to 87% reported that reviews have a significant influence on their purchases (Lipsman 2007). The consumers involved in this survey stated to be willing to pay up to 99% more for items that received very positive reviews. In addition, more than 30% of Internet users have posted a comment on a product or service.

The importance of user-generated opinions in decision making processes is clear. As Kannan et al. (2012) argue, due to the ever-growing amount of different products with similar characteristics, customers are often searching for authentic, user-generated reviews to estimate the actual utility value of products. Moreover, due to the ubiquity and invasiveness of online advertisement, consumers rely on user-generated opinions as a source of information about products and services (Chevalier and Mayzlin 2006). According to a recent study by Pang and Lee (2008), the attention that users pay to user-generated reviews also depends on the specific product being considered (see Table 1).

In many contexts, users cannot test the service in advance, and most of the available online information is

Sector	Customers
Hotels	87%
Travel	84%
Restaurant	79%
Legal	79%
Automotive	78%
Medical	76%
Home	73%

Table 1 Customers identifying online reviews as having a significant influence on their purchases in various economic sectors (source: comScore, Inc./The Kelsey Group).

generated by the producers for promotional purposes. Thus, a reasonable approach to obtaining unbiased information consists of reading opinions of other customers who previously tested the service. Analogously, service providers can adjust their business decisions based on users' opinions, promptly reacting to negative feedback (Chen and Qi 2011). Given the sheer volume of reviews generated every day, manual inspection is time-consuming at best, and often simply unfeasible. Hence, automatic sentiment detection offers a promising approach to analysing, summarizing, and aggregating opinions expressed in unconstrained natural language for wide audiences of consumers and service providers.

To date, most sentiment detection techniques have aimed at capturing the overall sentiment expressed in a review. However, as Moilanen and Pulman (2007) pointed out, the classification of the overall sentiment of a document can give misleading results. Atomic sentiment about single, distinct, product aspects can be overlooked. As a result, the overall sentiment cannot be simply derived by applying some algebraic operator to all of the feature polarities.

It is therefore beneficial to focus on *features*, i.e. aspects of a product or a service that can be rated independently in a review. For example, a review of an Mp3 player is likely to discuss distinct aspects like sound quality, battery life, user interface, and design, and a single product can trigger positive opinions about one feature, and negative opinions about another. Similarly, in hotel reviews, customers discuss specific features of their experience such as location, room size, staff's friendliness, hygiene, food quality, and availability of services. Different customers prioritize to different features. For instance, a couple on a honeymoon might not attribute huge importance to the hotel's Internet connection, whereas this aspect can be paramount to a corporate executive on a business trip (Titov and McDonald 2008).

In order to tackle this issue, we present a novel feature-based polarity analysis technique, which combines statistical techniques with natural language processing. First, we present a method for automatically detecting the salient features of a product, capturing the user's perspective. This analysis mines how much attention the user dedicates to each feature based on the term frequency, extracting domain knowledge in a bottom-up process. Second, for each review, we model the user's sentiment by estimating the degree of positive or negative polarity with respect to each salient feature. To extract such fine-grained sentiment information from raw text, we model each review as a set of sentences. A sentence is formalized as a syntactic dependency graph, used to analyze the semantic and syntactic dependencies between its terms, and identify the terms referring to features. To produce the mapping between lexical terms and features, we utilize a semantic similarity measure, taking into account synonyms and feature-related terms. Subsequently, by leveraging the sentence formalization as dependency graphs, we estimate the local polarity degrees with respect to the features identifying the terms that express a non-neutral opinion.

The remainder of this paper is organised as follows: Section 2 surveys related research, focusing on the works that integrate natural language pre-processing techniques with the usage of structured domain information. Section 3 presents our feature-based polarity detection technique, which integrates syntactic and semantic analysis of raw text to extract sentiment polarities at the feature level. Section 4 considers a real-world scenario, applying our approach to a corpus of hotel reviews from TripAdvisor.[1] Empirical evidence collected through experiments supports the effectiveness of our approach. To ease the empirical comparison of our approach with alternative approaches, we have made the full evaluation dataset available online.[2] Finally, Section 5 draws conclusions about the work presented in this paper, discussing limitations and future research directions.

2 Related work

The automatic detection and categorization of sentiment expressed in natural language constitutes a remarkable research challenge, and has generated a nexus of techniques and approaches. In this context, our approach offers a complete framework for unsupervised feature-based sentiment analysis, starting from a raw text corpus and returning a set of opinionated features. To achieve this result, we tackle a number of specific natural language processing problems. This section gives an overview of the large body of work that informs our approach.

[1] `http://www.tripadvisor.com`
[2] `http://github.com/ucd-spatial/Datasets`

2.1 Overall sentiment estimation

Pioneering work on sentiment analysis has been conducted by Pang et al. (2002), comparing different machine learning approaches such as Maximum Entropy, Support Vector Machines, and Naive Bayes, to classify sentiment in movie reviews. This study reported poor performances for all of these methods, stressing the need for a deeper understanding of the linguistic and syntactic aspects of sentiment in natural language. More recently, Pang and Lee (2008) and Liu (2012) have conducted extensive surveys of the open challenges in this research area. Opinion mining techniques have been devised and evaluated on many domains, including news stories (Godbole et al. 2007), films (Annett and Kondrak 2008; Zhou and Chaovalit 2008), electronic gadgets (Hu and Liu 2004b; Titov and McDonald 2008), and hotels (Pekar and Ou 2008; Ye et al. 2009; O'Connor 2010).

Most of the works in this area focus on the categorization of *overall* sentiment, capturing a form of average polarity at the document level (Turney 2002; Beineke et al. 2004; Hiroshi et al. 2004; Pang and Lee 2004). In this context, Morinaga et al. (2002) assign a sentiment degree to each word relying on term frequencies, and statistical models of favorability, i.e. whether the expressions indicate positive or negative opinions. However, as Moilanen and Pulman (2007) point out, such statistical sentiment classifiers appear to perform well with sufficiently large text corpora, but fail to handle smaller sub-sentential units, such as compound words and individual phrases.

2.2 Syntax and sentiment aggregation

A number of researchers have applied natural language processing (NLP) techniques to detect features in small chunks of text (Nadeau and Sekine 2007; Holz and Teresniak 2010; Missen et al. 2012). Hatzivassiloglou and McKeown (1997) use textual conjunctions to separate words with similar or opposite sentiment. Similarly, Matsumoto et al. (2005) observe that the order and the syntactic relations between words are extremely important to support sentiment classification. Hence, they construct dependency trees for each sentence, and prune them to obtain subtrees to evaluate linguistic patterns. Our approach bears parallels with this body of work, but we use POS tagging and dependency trees to capture the relationships existing among syntactic elements, instead of using them for pattern analysis.

Another crucial aspect to develop effective computational models of sentiment is the identification of rules to combine small units into larger semantic entities. In natural language, the sentiment conveyed by atomic constituents is aggregated in larger units through complex compositional mechanisms. Moilanen and Pulman (2007) provide a technique to compute the polarity of grammatical structures, outlining a framework for sentiment propagation, polarity reversal, and polarity conflict resolution.

2.3 Feature detection and clustering

Although most approaches aim at extracting overall polarities, efforts have been undertaken to recognize the sentiment at the feature level (e.g. Popescu and Etzioni 2005). Such techniques typically consist of two steps: (i) identifying and extracting features of an object, topic or event from each review, sentence, or document, and (ii) determining the opinion polarity of the features. Hu and Liu (2004a,b) have devised an approach to generate feature-based summaries of online reviews, aiming to detect salient *opinion features* that users tend to either like or dislike. Along similar lines, Titov and McDonald (2008) propose a multi-grain approach to extract opinion features, extending topic detection techniques such as Probabilistic Latent Semantic Analysis (pLSA) and Latent Dirichlet Allocation (LDA).

From a different perspective, Zhai et al. (2011a) have developed a semi-supervised technique to identify clusters of features, i.e. sets of synonyms that are likely to refer to the same product features. Their technique needs a fixed number of clusters, and a starting set of features to bootstrap the process. The same task can be performed using a topic model such as LDA (Zhai et al. 2011b). Unlike these clustering approaches, our feature extraction procedure, presented in Section 3.1, does not look for feature clusters, but extracts individual features in an unpervised procedure, and determines automatically a suitable number of salient features.

2.4 Co-reference resolution

Another important component to extract user opinion from raw text is the co-reference of a feature and its opinionated attributes. When analysing the sentence "I bought this camera yesterday from an online shop, it's really compact and light.," precious information is lost if the approach does not consider that pronoun *it* refers to the camera and not the shop. A supervised machine learning approach to co-reference to has been presented by Ding and Liu (2010). Their system learns a function to predict whether a pair of nouns is co-referent, building coreference chains based on feature

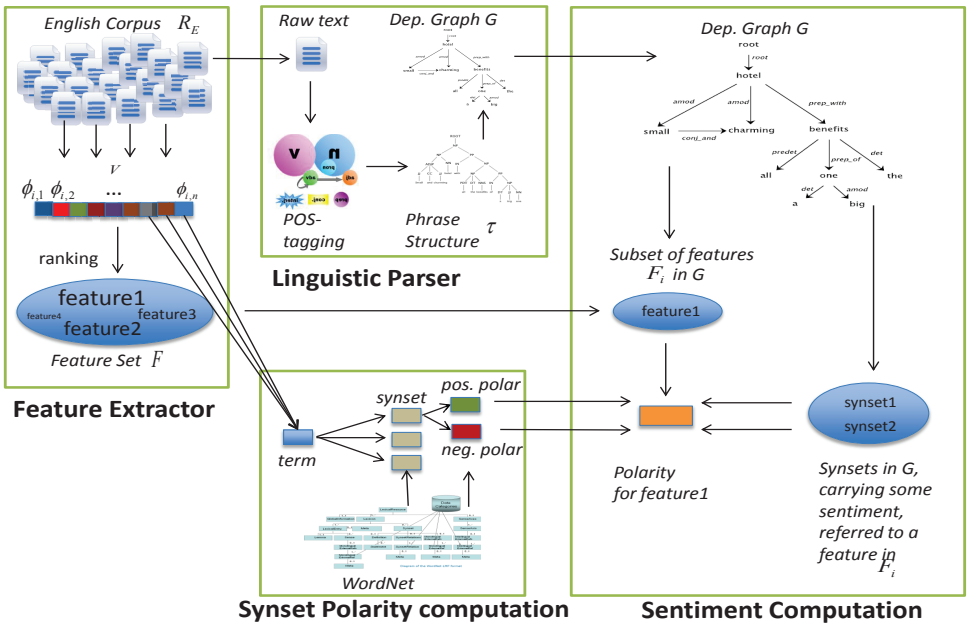

Fig. 1 Flowchart of the proposed hybrid model for detecting feature-based polarities.

vectors that model a variety of contextual information about the nouns.

Tackling the co-reference problem, the double propagation method extracts attributes referred to objects iteratively, given a opinion lexicon as seed (Qiu et al. 2009, 2011). Double propagation assumes that features are nouns and opinionated attributes are adjectives, and that they tend to be associated in the corpus by certain given rules. Hence, attributes can be identified from known features, and features can be detected by known attributes, propagating the detection back and forth from attributes to features and vice-versa. The propagation process ends when no more opinionated attributes or features can be found in the corpus. Variants of double propagation have been developed specifically for small and large corpora (Zhang et al. 2010). In our approach, we propose an alternative unsupervised method to co-reference resolution based on dependency graphs (see Section 3.2).

2.5 Sentiment visualization

Once a sentiment model has been generated, several approaches explore the visualization of the outcome of automatic sentiment analyses, and enable the intuitive exploration and inspection. The system devised by Liu et al. (2005) enables the visual comparison of different products with respect to the amount of positive and negative reviews on different product features. Oelke et al. (2009) propose a scalable alternative in order to aggregate large numbers of products and features, clustering similar users. Similarly, the system by Miao et al. (2009) visualizes the sentiment expressed in product reviews over time. Positive and negative opinions were therefore aggregated over time and displayed with different charts. Recently, Wu et al. (2010) propose a novel visual analysis tool, called OpinionSeer, for user-generated reviews where uncertain sentiment through time is visually represented and aggregated. Unlike Miao et al. (2009), this approach takes the temporal dimension into account.

3 A hybrid approach to computing feature-based polarities in reviews

In this section, we describe our unsupervised four-step technique to detect fine-grained opinion in user-generated reviews. Unlike existing approaches that estimate the overall polarity of a review, we detect the local polarities expressed about the salient features of a considered domain. A polarity is a real number that quantifies the user's positive, neutral, or negative opinion about a feature. For example, a positive polarity can be assigned to the word "clean," a neutral polarity to "red," and a negative polarity to "horrible."

Given a raw text corpus of reviews, as we focus on the subset of reviews written in English, we first need to automatically detect the language of the reviews. The language detection is performed via a text categorization approach based on the similarity between

Symbol	Description
ϕ	Augmented normalized frequency of a term
F	Set of salient features
δ	Drop in ϕ between two terms
f	Feature $\in F$
R	Corpus of reviews in English
r	Review $\in R$, set of sentences
V	Vocabulary utilized in R
S	Sentence $\in r$, set of w
w	Word $\in S$
τ	Phrase structure tree
G	Dependency graph for S
e	Edge in dependency graph G
sim	Semantic similarity function $\in [0, 1]$
W	WordNet (set of synsets)
s	Synset $s \in W$
pos_s	Positive polarity for synset s
neg_s	Negative polarity for synset s
T_f	Set of terms referring to feature f
$pol_{f,r}$	Polarity of feature f in review r
$syns$	Word sense disambiguation function
σ	Agreement threshold $\in [0, 1]$

Table 2 Notations

the reviews and a set of language template vectors.[3] The reviews are represented as term vectors, weighted with TF-IDF weights, and compared against a set of language template vectors, which consist of weighted vectors containing the most frequent words of a language. The cosine vector similarity measure is utilized to detect the language of each review, which reaches very high precision (Baldwin and Lui 2010). Reviews in languages other than English are discarded.

The approach starts with a statistical analysis of the corpus to extract the most frequent domain-related features. From the user's point of view, these features represent the most salient aspects of the product discussed in the reviews, such as the "weight" of a laptop, or the "location" of a hotel (Section 3.1). Subsequently, we model each review as a set of dependency graphs, which are populated with lexical nodes representing the terms, encoding their syntactic and semantic relations (Section 3.2). The dependency graphs are then utilized to detect the terms referring to a feature, which expresses some non-neutral opinion, including compound expressions, e.g. "very convenient location." In this phase, a SentiWordNet-like approach is used as a source of polarity values (Section 3.3). Finally, the polarities of terms are aggregated into the feature polarities (Section 3.4). The structure of the proposed approach is outlined in Figure 1. All the notations utilized in this article are summarized in Table 2.

[3] This step is performed with the jExSLI tool, available at `http://hlt.fbk.eu/en/technology/jExSLI`

3.1 Step I: Extraction of salient features from the review corpus

The domain in which the polarities have to be detected is represented by a corpus of text reviews $R = r_1, r_2, \cdots, r_n$. Instead of computing the overall polarity of each review r_i, we aim at identifying the most characteristic features of the domain R, and retrieve the polarities expressed by the users about each feature. The manual construction of a list of salient features is time-consuming, and does not necessarily reflect the features considered salient by the reviews' authors. Hence, we present a method for the automatic detection of the most salient features of a product, from the user perspective, using a statistical technique. This analysis quantifies how much attention the users dedicate to each feature based on the statistical distribution of terms in the corpus.

In order to construct dependency graphs, the parts of speech (POS) in the raw text have to be identified. The identification of nouns, adjectives, verbs, and other parts of speech is crucial to the successful modelling of feature-level sentiment. The reviews are therefore POS-tagged, and common stop words are eliminated. At this point, given the subset of reviews in English R, the set of salient features F has to be selected. Although the inverse document frequency (IDF) approach might look appropriate at first glance, it is not effective in this context. IDF lowers the weight of terms that occur very frequently in the corpus, while it increases the weight of terms that occur rarely. In the context of feature extraction, terms that appear in most documents tend to be highly relevant for the domain. For example, terms like 'room,' 'bathroom,' and 'breakfast' occur in nearly every hotel review, and are indeed salient features.

Hence, F is computed as the set of noun terms having the highest augmented normalized term frequency, as defined by Salton and Buckley (1988). Formally, given a noun word $w_j \in V$, where V is the English vocabulary of R, we calculate its augmented normalized frequency ϕ_j as follows:

$$\phi_j = \sum_{i=1}^{|R|} \left(.5 + .5 \frac{tf_{j,i}}{tf_i^{max}} \right)$$

where $tf_{j,i}$ is the term frequency value of the term j in the review i, and tf_i^{max} represents the highest term frequency value in the review i. By increasing the minimum frequency value to .5, the augmented normalized approach tends to preserve infrequent words that could be relevant in the domain, reducing the gap between the most and the least frequent words.

Therefore, we select as relevant features F the set of terms having the highest augmented normalized fre-

Term	Frequency ϕ	Rank
room	121,746.99	1
staff	42,441.89	2
location	29,345.32	3
breakfast	27,080.25	4
place	20,684.92	5
service	19,001.75	6
bathroom	17,471.16	7
restaurant	15,459.13	8
area	12,334.79	9
view	11,247.15	10

Table 3 The top ten hotel features, their frequency ϕ, and rank extracted from the corpus of hotel reviews described in Section 4.1.

quency ϕ, above a dynamic threshold. The manual selection of a frequency threshold would fail to capture the actual salience of the features, introducing an arbitrary bias in F. Thus, in order to select the salient features F, we leverage a fully automatic ranking model that dynamically identifies a *critical drop* $\hat{\delta}$. The underlying intuition is that the frequency of salient features must be higher than the average frequency of the most discussed terms. The parameter z is the minimum cardinality of set F, where $z > 0$. Hence, the ranking model selects F in a five-step process:

1. The system ranks the n terms in descending order of augmented normalized frequency value ϕ, so that $\forall i,\ \phi_{i+1} \leq \phi_i$;
2. For each pair ϕ_i and ϕ_{i+1}, the drop δ_i is defined as $\phi_i - \phi_{i+1}$ (with $i \geq z$). The *maximum drop* δ_k^{max} is therefore identified between the terms k and $k+1$, so that $\forall i \neq k,\ \delta_k^{max} \geq \delta_i$;
3. The *average drop* $\bar{\delta}$ is computed as the average of all the terms having ϕ higher than the ϕ_{k-1}, i.e. the terms ranked higher than the terms having maximum drop;
4. The first drop which is higher than the average drop $\bar{\delta}$ is selected as the *critical drop* $\hat{\delta}$;
5. The terms ranked higher than the critical drop $\hat{\delta}$ are selected as F ($w_j \in F \Rightarrow z < j < k-1$).

If the maximum drop is detected between the first and the second entry and $z \leq 1$, the resulting F would only contain one noun. In most cases, this state of affairs might not be desirable, and z must be set to an appropriate value (e.g. $z = 5$ in the evaluation in Section 4). The resulting subset of terms F contains salient features in the users' eyes. Table 3 reports an example of F extracted from a real corpus of hotel reviews (see Section 4.1).

3.2 Step II: Identification of dependency relations in user-generated reviews

Dependency grammars have been developed in both theoretical and computational linguistics for syntactic representation of sentences. They have attracted interest because of their simplicity, efficiency and manageability (McDonald and Nivre 2011). In dependency grammars, the content of a sentence and the relations between its terms are represented and formalized through binary, asymmetrical relations called *dependency relations* or *dependencies*. A dependency holds between a governor word, the *head*, and a syntactically subordinated element, called the *dependent*. The resulting structure is therefore a syntactic dependency graph, where the nodes represent lexical elements, and the directed edges represent grammatical relations pointing from a governor to a dependent.

Formally, a sentence S is an ordered vector of terms $S = \{w_0, w_1 ... w_m\}$, where the order represents the original position of each term within the sentence. The sentence s can be represented as a dependency graph G. The dependency graph is a labeled directed graph $G = (V, E, l)$, where V is the set of nodes representing the lexical elements w_i, E the set of edges (i.e. dependency relations) among the nodes. The function l is a labeling function on E which defines the type of dependency between two lexical elements w. We use the notation $e_{i,j} : w_i \rightarrow w_j$ to state that there exists an edge in G which connects w_i (the head) to w_j (the dependent), where $w_i, w_j \in V$ and $e_{i,j} \in E$. Furthermore, $w_i \rightarrow^* w_j$ denotes a reflexive transitive closure of an arc relation. G is characterized by the following properties:

- *Single-head*: each node has at most one head $(w_i \rightarrow w_j \wedge w_k \rightarrow w_j) \Rightarrow w_i = w_k$;
- *Acyclicity*: graph does not contain cycles $w_i \rightarrow w_j \Rightarrow \neg(w_j \rightarrow^* w_i)$.

The mapping of each review, viewed as a set of sentences, to its syntactic representation (i.e. a dependency graph), is performed through a three-step parsing algorithm. The first step consists of the POS tagging, achieved by training a tagging model on the annotated corpus proposed by Marcus et al. (1993) and therefore by calculating the probability $p(t_j|w_i)$ of assigning a tag t_j to the term w_i using a maximum-likelihood estimation (as in Klein and Manning 2003). At this point, the POS-tagged sentence takes the form of an ordered set $s_{tag} = \{w_0/t_0 ... w_n/t_n\}$. For example, the sentence "Small and charming hotel with all the benefits of a big one" results in the following POS-tagged representation:

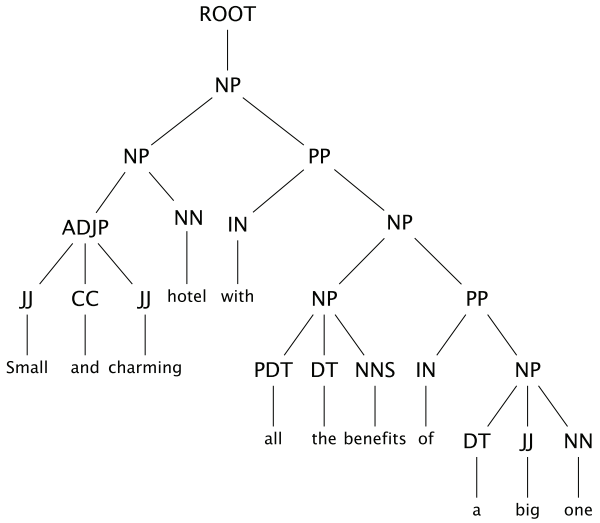

Fig. 2 Example of phrase structure for the sentence "*Small and charming hotel with all the benefits of a big one.*"

Small/JJ and/CC charming/JJ hotel/NN with/IN all/PDT the/DT benefits/NNS of/IN a/DT big/JJ one/NN.[4]

The second step concerns the construction of a phrase structure tree τ_S by applying the CKY algorithm of Kasami (1965) to the POS-tagged sentence. Syntactic phrase structure trees, also known as constituent trees, are representations of sentences where terms are grouped in syntactic units. In these trees, words that are syntactically related in the sentence are grouped into units, called phrases, or constituents. The leaves of the tree represent the original sentence terms, whereas internal nodes are labeled with non-terminal symbols, i.e. POS-tags at the lowest internal level, and then phrases at all of the other levels. These trees are generated by using a set of phrase structure rules. For example, the noun phrase (NP) "a big one" composed by a determiner ("a"), an adjective ("big") and a noun ("one") is represented through the rule $NP \rightarrow DT\ JJ\ NN$. Figure 2 shows the entire τ-structure derived from the aforementioned example sentence.

Subsequently, in the third step, we transform the phrase structure tree τ_S in order to produce the final dependency graph G. This is achieved through a bottom-up approach performed on the τ-structure where each phrase is analyzed with the goal of identifying the head and its dependents. The heads are identified by following a set of linguistic rules established by Collins (1999). In this process, the priority is given to semantics rather than pure syntax.[5] In general, heads are mainly content words, while grammatical elements, such as auxiliaries and complementizers, end up being their dependents. Each dependency $w_i \rightarrow w_j$ connecting a head w_i to its component w_j, is then labeled with the type of grammar relation that guided its identification.

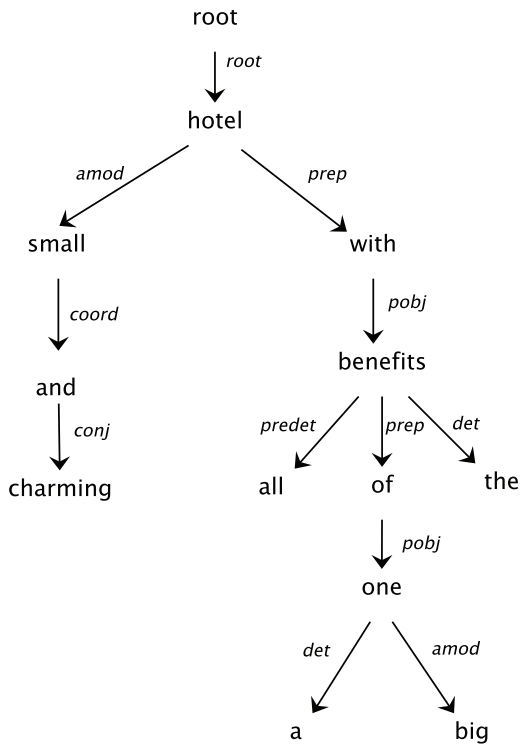

Fig. 3 Dependencies graph inferred from the phrase structure tree shown in Figure 2.

Figure 3 shows the dependency graph corresponding to the example sentence. In the noun phrase (NP) "a big one," the noun "one" is identified as the syntactic head while the elements "a" and "big" are identified as its dependents. In particular, the element "a" is then identified as the determiner of the element "one," while "big" is connected to the head through an "adjectival modifier" relation. This process is recursively carried out on the phrase structure tree τ_S from the leaves to the root of the original τ-structure, until the head of the highest phrase is found and all the lexical elements have been connected.

Finally, in order to preserve the semantics of the sentence, the dependency graph is collapsed by linking the terms that are connected through a preposition or conjunction. We select the pairs of terms w_i and w_j that are not directly connected, i.e. $(w_i, w_j) \notin E$. If a pair is

[4] Where *JJ* means adjective, *CC* coordinating conjunction, *NN* noun, *IN* preposition, *PDT* predeterminer, and *DT* determiner. A complete list of the categories has been defined by Marcus et al. (1993).

[5] In some sentences, the semantic and syntactic representation may not correspond. For a detailed discussion, see De Marneffe et al. (2006).

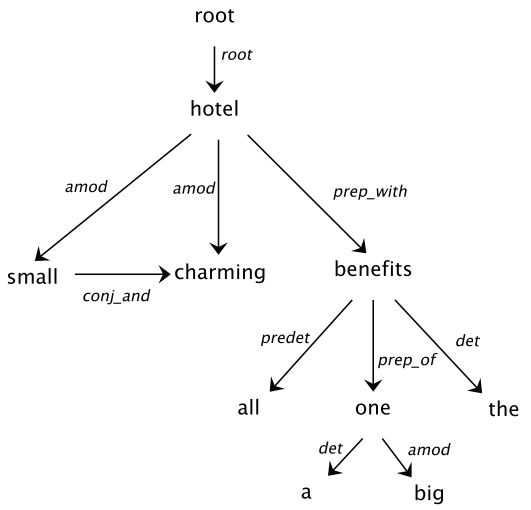

Fig. 4 Collapsed dependency graph inferred from the dependency graph shown in Figure 3.

connected through another syntactic element w_k that has been POS-tagged as a preposition or conjunction, i.e. $(w_i, w_k), (w_k, w_j) \in E$, w_i and w_j can be directly connected through a dependency relation.

An example of a collapsed dependency graph is shown in Figure 4. In the non-collapsed representation the adjective "charming" is dependent on the element "and," while both "charming" and "small" can be easily interpreted as adjectival modifiers of the element "hotel." By contrast, the collapsed representation preserves the original semantics of the two original adjectival modifier relations (*amod*) between the head "hotel" and its children. Note that the coordination is maintained with the conjunction relation (*conj_and*) between the two considered adjectives.

3.3 Step III: Calculation of statistical polarities of WordNet synsets

The determination of a polarity score for a term is a central task in our technique. In order to identify the sentiment at the feature level, we need to estimate the degree of positivity or negativity that each term carries in each review. This problem is complex because the terms in unconstrained natural language are often semantically ambiguous, and their meaning is context-dependent. For this reason, as in the lexical database WordNet (Fellbaum 1998), instead of computing a high level polarity of each vocabulary term, we estimate the polarity for each different meaning of a term ('word sense' in WordNet). Terms having similar meaning are grouped in "synsets." The main intuition behind this approach is that different meanings of the same term may convey different sentiments.

As a source of sentiment polarities, we utilize a SentiWordNet-like technique, which attributes polarity values to each WordNet synset (Baccianella et al. 2010). In order to assign polarities to each term in a review, the terms in raw text have to be mapped to the corresponding WordNet synset s_i. This step, as explained in Section 3.4, is performed through the word sense disambiguation technique by Pedersen and Kolhatkar (2009). It is important to note that the same synset can assume a *negative* or *positive* polarity, depending on the context. Each term, mapped to a single WordNet synset s_i, is then associated to two numerical scores pos_{s_i} and neg_{s_i} which indicate how positive and negative the synset can be. We calculate the polarities of each synset by using a two-step algorithm:

1. A semi-supervised learning step in which polarity values are assigned to two sets of *seed nodes*. This set consists of two subsets; one subset of "paradigmatically positive" synsets and another one consisting of "paradigmatically negative" synsets (Turney and Littman 2003). The polarities are then propagated automatically to other synsets of the WordNet graph by traversing selected semantic relations. For relations such as "see-also" and "synonymy," the polarity sign is preserved, whilst for "antonymy" relation the polarity sign is inverted. This propagation is performed within the minimal radius that guarantees no conflicts among the relations, that is, until a node labeled as positive points to a node already linked to some negative seed, or vice-versa). In other words, we only propagate the polarities to the nodes that are univocally connected to a positive or a negative seed.
2. A random-walk step is executed on the whole WordNet graph starting from the seed nodes, and iteratively propagates the pos_{s_i} and neg_{s_i} to all of the synsets. This approach preserves or inverts the polarity of each node based on the number of positive and negative relations that connect it to the seeds. The process ends when a convergence condition is reached. This condition is satisfied when all the nodes have maintained the same polarity sign (positive or negative) after two consecutive steps.

At the end of these two steps, each synset $s_i \in W$, where W is the set of synsets extracted from WordNet, is associated with two values pos_{s_i} and neg_{s_i}, both in interval $[0, 1]$. Note that the sum of pos_{s_i} and neg_{s_i} is not necessarily equal to 1. Each term in V can now be associated to a sense in W through a word sense disambiguation technique (e.g. Pedersen and Kolhatkar

Term	WordNet Synset	Pos. Pol. pos_{s_i}	Neg. Pol. neg_{s_i}
small	small#a#1	0	0.375
and	and#CL	-	-
charming	charming#a#1	0.875	0
hotel	hotel#n#1	-	-
with	with#r#ND	-	-
all	all#CL	-	-
the	the#CL	-	-
benefit	benefit#n#2	-	-
of	of#r#ND	-	-
a	a#CL	-	-
big	big#a#1	0.25	0.125
one	one#n#1	-	-

Table 4 Polarity estimation of the sentence "small and charming hotel with all the benefits of a big one."

2009), and then associated to a positive and a negative polarity.

As an example, let us consider again the sentence "small and charming hotel with all the benefits of a big one." The sentence is first analyzed in order to detect the intended meaning of each term (i.e. the specific WordNet synset). Subsequently, for each synset, we retrieve the positive and negative polarities through the two-step process above. The resulting polarities are shown in Table 4. As is possible to notice, among the six possible different meanings of "charming," the system associates the term with the first adjectival WordNet synset, charming#a#1, defined as "endowed with charming manners; a charming little cottage; a charming personality."

Among all terms, only the adjectives "small," "charming" and "big" are identified as terms carrying some non-neutral sentiment. Terms such as "big" have both a positive and negative polarities. Such terms do not convey a positive/negative sentiment in isolation: their polarity strongly depends on associated modifiers (adverbial modifiers, negations, and/or combination of adjective and nouns).

3.4 Step IV: Computation of feature polarities

The terms expressing some non-neutral opinion about the features can be used to estimate the features' polarities. To reach this goal, our approach exploits the dependency properties of the collapsed dependency graph (see Section 3.2), and the polarity of terms calculated in the previous section.

First, given a review $r_i \in R$ formalized as a set of collapsed graphs, we search for the terms that refer to the salient features F. In order to detect these terms in the collapsed graphs, we do not only consider exact lexical matches, but we also include synonyms, accounting for morphological variations and similarities. This is important to cope with the high linguistic diversity that characterizes online reviews. The synonyms are identified by using a semantic similarity function.

More specifically, we adopt the approach to semantic similarity proposed by Ponzetto and Strube (2007), which estimates the similarity of terms based on statistically significant co-occurrences in large corpora, such as Wikipedia. Given a pair of terms in input, they retrieve Wikipedia articles referring to the terms, and compute the semantic similarity between the two terms based on the paths that connect these articles in the category graph.

Hence, we calculate the similarity between a noun term in the review and a term feature in F with a function sim that returns a similarity score in interval $[0, 1]$. If a term is not present in the corpus, its similarity with other terms is set to 0. On the other hand, if the two terms are identical, their similarity is 1.Using this similarity evaluation approach, we are able to estimate the semantic similarity between the terms in the review and the feature terms. This task is carried out in a three-step process:

1. For each term feature $f \in F$, we compute its semantic similarity with each term w_j in r_i, and select the terms whose similarity is above a given threshold. The resulting subset of features F_i represents the features that are discussed in the review r_i.
2. In the collapsed dependency graphs for r_i, we detect all the terms w_k which refer to some features $f \in F_i$, and have a non-neutral polarity (i.e. $pos_{w_k} > 0 \vee neg_{w_k} > 0$). For each of these terms w_k, we retrieve the closest connected noun in the collapsed dependency graph, and check if it is associated to one of the features $f \in F_i$ through the similarity measure sim. If this is the case, we consider the sentiment value carried by w_k to determine the overall polarity expressed in r_i about the feature $f \in F_i$. At the end of this step, each feature $f \in F_i$ in r_i has a subset T_f of non-neutral terms.
3. For each detected feature $f \in F_i$, we analyze the collapsed graphs in order to detect if a term w_x in T_f is directly connected to a term w_y which is labeled as adverbial modifiers, negations, or both. If this is the case, we concatenate w_y to w_x within T_f.

In the example discussed above, Table 4 showed three terms having non-neutral polarities ("small," "charming," and "big"). The system tries to detect the terms they refer to by analyzing the collapsed dependency graph shown in Figure 4. The syntactic element closest to the terms "small" and "charming" is the noun "hotel," which is therefore estimated as their target el-

ement. In the collapsed graph, their distance w.r.t. the term "hotel" is 1. Similarly, the noun "one" is the target term for "big."

In the last step, we detect not only non-neutral terms, but also adverbial modifiers, negations, and combinations of both that can radically alter the meaning of terms. For example, our approach captures the adverbial modifier in "incredibly charming hotel" as $advmod[charming, incredibly]$, and the negation in "the hotel is not charming" as $neg[charming, not]$. The approach also handles complex combinations of both ("the hotel is not too small": $neg[small, not], advmod[small, too]$). The order of the dependencies is fundamental to this process.

A subset of features $F_i \in F$ has been identified for each review $r_i \in R$. For each feature $f \in F_i$, the corresponding set of lexical components T_f captures all the non-neutral terms, including adverbial modifiers and negations. Thus, we define a word sense disambiguation function $syns(t, r_i)$ that returns the WordNet synset related to the term t in the context of the review r_i, executed with the tool devised by Pedersen and Kolhatkar (2009). As shown in Table 4, each term t is associated with a specific WordNet synset. Therefore, by leveraging the approach described in Section 3.3, we can estimate a positive and a negative polarity for each feature. The local polarity pol_{f,r_i} of a feature $f \in F_i$ in the review r_i can be computed as:

$$pol_{f,r_i} = \sum_{t \in T_f} \frac{pos_{syns(t,r_i)} - neg_{syns(t,r_i)}}{|T_f|}$$

The computed polarities take into account both the linguistic representation of the sentence through the dependency graph, and the statistical estimation of sentiment by using a preliminary computed sentiment value, based on WordNet. In the next section, this approach is evaluated in a real-world scenario.

4 Evaluation

In this section, our approach to opinion mining is evaluated on a real world corpus of hotel reviews, originally collected by Ganesan and Zhai (2012). In particular, we consider a corpus of $\approx$ 259,000 user-generated reviews, containing full hotel reviews collected from TripAdvisor. Note that no pre-processing, such as stop words elimination or stemming, was performed on the corpus. The original raw text is necessary for the construction of the dependency graphs and the linguistic analysis of the reviews.

The aim of this experimental evaluation is twofold. First, we explain the impact of the parameters utilized in the presented technique. Second, we assess the machine performance against a human-generated ground truth, collecting feature polarities about a random sample of reviews. The machine-generated polarities are thoroughly compared with the human dataset, obtaining promising results. Finally, we discuss the advantages and disadvantages of the proposed approach, highlighting directions for future work. The next sections describe the experiment design (Section 4.1), the validation of the ground truth (Section 4.2), polarity recall (Section 4.3), the polarity sign precision (Section 4.4) and polarity degree precision (Section 4.5).

4.1 Experiment design

In order to assess the proposed approach to sentiment analysis, we designed a user evaluation. In this experiment, human subjects are asked to read and interpret real hotel reviews, rating the author's sentiment about specific features. The human subjects perform the same sentiment analysis that our system performs automatically.

First, we randomly selected 14 hotel reviews from the corpus of hotel reviews by Ganesan and Zhai (2012). The 14 reviews were processed with the system. Among the features extracted automatically, up to four random features were assigned to each review, for a total of 40 features (19 unique features in total).[6] In total, the corpus contained 185 sentences (with an average of 13.2 sentences per review, in range [6,22]).

In order to keep the task within a reasonable length and to analyze the reliability of our test, we split the set of 14 reviews into two questionnaires, A and B, each containing eight reviews. One review was present in both questionnaires, and is used as control question to verify the reliability of the groups. To collect the answers, two online questionnaires were created. The two questionnaires were disseminated on August 28, 2012. The human subjects were randomly assigned to either questionnaire A or B. Eventually, 23 subjects took questionnaires A, and 19 questionnaire B, for a total of 42 responses. Three responses were incomplete and were therefore discarded, for a total of 39 valid responses (21 in A, 18 in B).

The polarity rating was expressed on a 4-point Likert scale, with a "not available" option in case the feature was not discussed in the target review. The four points were labelled as "very negative," "negative,"

[6] In total, the system detected 33 features for the considered domain. The 19 unique features randomly selected for the experimental evaluation are: *room, staff, location, breakfast, place, service, bathroom, restaurant, area, desk, view, shower, bed, pool, city, Internet, reception, rate, parking.*

"positive," and "very positive." An alternative would have consisted of a 5-point Likert scale, with a middle point (e.g. "neutral"). However, in the domain of online reviews, perfectly neutral judgements are rare, as the users tend to express some degree of satisfaction or dissatisfaction about a specific feature. We chose not to include the middle point to reduce the central tendency bias, forcing subjects to choose between a positive and a negative opinion, even in cases where the polarity is very weak. The bias that this choice introduces in the results is debatable, and there are no conclusive findings to prefer one solution over the other (Dawes 2008). The entire dataset, including the reviews and the human responses, is available online under an Open Knowledge license.[7]

4.2 Human dataset validation

The polarity judgements collected from human subjects through the questionnaires A and B provide a psychological ground truth against which the automatic technique can be evaluated. In order to assess the validity of the collected data, it is essential to measure the inter-rater agreement (IRA), i.e. the extent of agreement among human raters on the polarities expressed about the target objects (Banerjee et al. 1999). For this purpose, we make use of Fleiss' kappa, an index tailored to capture the IRA among a fixed number of raters on categorical ratings (Fleiss 1971).

Fleiss' kappa (κ) indicates the extent of agreement among raters in the interval $[0, 1]$, in which 0 means that the human subjects gave random ratings, and 1 means that they gave exactly the same ratings to all the target objects. For the two questionnaires A and B, the average κ is .65, and bears high statistical significance ($p < .0001$). Considering the high subjectivity of polarity judgments, we deem this agreement among the subjects to be satisfactory.

Another salient aspect to observe is the distribution of polarities in the dataset. Figure 5 shows the distribution of polarities selected by the human subjects, confirming the homogeneity of the results across the two questionnaires. All polarities are represented in the questionnaires. In the set of 40 features on 14 random reviews, positive polarities are the majority of judgments (57.6%), the rest being constituted by negative polarities (14.2%), and "not available" features (28.2%).

The distribution of responses to questionnaire A and B indicate no significant difference between the two

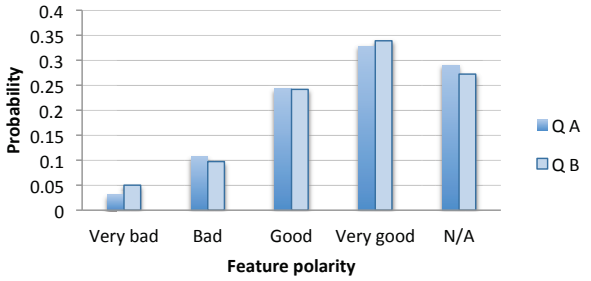

Fig. 5 Distribution of polarities in the responses in questionnaires A ('Q A') and B ('Q B').

groups of subjects ($p > .1$), Furthermore, the substantial homogeneity of the two groups can be observed in relation to the review shared among the questionnaires, used as a control variable. The control review received consistent polarity judgments on its three features (respectively "service," "room," and "Internet"), resulting in almost perfect agreement ($\kappa = .82$). Hence, this human-generated dataset can be used as a ground truth to evaluate the proposed computational approach.

4.3 Polarity recall

A key aspect of the presented approach is the automatic extraction and detection of opinionated features. To assess the system's ability to distinguish between opinionated and non-opinionated features, we quantify how many features have been estimated as opinionated by both the human subjects and the system, and how many features have been labeled by both as "not available." As Hiroshi et al. (2004) proposed, we define this measure as *polarity recall*, in interval $[0, 1]$. Using the polarity recall, we assess whether our system detects the same opinionated features as the human subjects, and we analyze when and why differences arise. To compute the polarity recall, we took several aspects into account. Although the subjects show a high IRA (see Section 4.2), the responses can significantly differ because of psychological and linguistic factors. The ambiguity of natural language, the reviewer's and subject's knowledge of the English language, the interpretation of the Likert scale, and the attention devoted to the test can have a considerable impact on the final result.

Hence, we introduce an agreement threshold σ in interval $[0, 1]$, which selectively discards responses discordant from the system. If the tolerance threshold σ is equal to 0, a total agreement is necessary to consider a feature as concordant with the system (100% of subjects). If $\sigma = 0.1$, the system's result is considered correct if at least 90% of the human subjects agree with the system (a maximum of 10% of discordant replies can

[7] `http://github.com/ucd-spatial/Datasets`

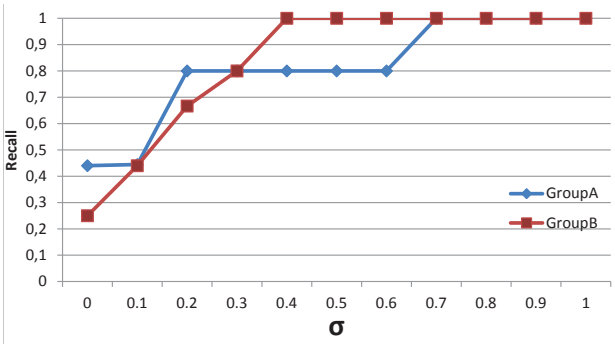

Fig. 6 Recall of the system in distinguishing between opinionated and non-opinionated features, based on the agreement threshold σ.

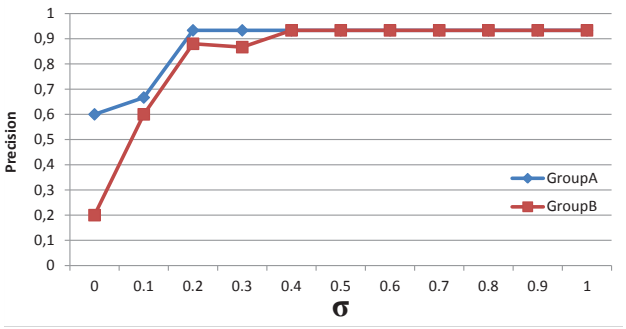

Fig. 7 Sign precision of the proposed system in detecting the feature polarities in the reviews, based on the agreement threshold σ.

be discarded). At the other end of the spectrum, $\sigma = 1$ implies that the system's response is correct if any subject has expressed the same opinion – potentially, all the responses but one can be discarded. To observe the system recall taking subject disagreement into account, we compared the human and system polarity by varying the tolerance threshold on the subjects' responses.

The polarity recall is shown in Figure 6. As is possible to notice, the recall is higher than 0.83 when discarding the 30% of discordant human subjects ($\sigma = 0.3$). In other words, for each question, discarding the 30% of the responses which are discordant with the average reply, our system retrieves the same opinionated feature detected by the users in approximately 80% of the cases. When $\sigma = 0$, even a single subject that identifies the presence of an opinionated feature incorrectly introduces considerable noise in the final results, resulting in lower recall. Both questionnaires A and B obtain consistent recall. Taking into account the disagreement in the responses with the threshold σ, the aforementioned recall values prove the efficacy of the proposed approach in detecting when a feature results opinionated within a considered product review.

4.4 Polarity sign precision

In order to assess the system's performance, we compute the system precision by comparing the machine-generated polarity values against those provided by the human subjects. The categorical polarities reported by the users, ranging from "very negative" to "very positive," were normalized to either -1 or 1, where -1 means negative and 1 positive. In this context, we define the *polarity sign precision* as the ratio between the cases in which the system and the human subjects assigned the same polarity sign to a feature (either negative or positive), and the total number of cases. The polarity precision falls in interval $[0, 1]$, where 1 indicates perfect agreement between the system and the human subjects. This measure focuses on the polarity sign, and not on the polarity degree, i.e. "bad" and "very bad" are both normalised to -1. A fine-grained assessment of the polarity degree is reported in Section 4.5.

As in the previous experiment, we introduce and vary a threshold σ in order to select and discard the responses that appear inconsistent with the other responses. The system's precision, computed for both questionnaires, is shown in Figure 7. As is possible to notice, our system is effective at capturing the polarity sign (positive or negative) about the salient features. The system obtains the same polarity values of the absolute majority of the users by only discarding the 10% of discordant responses ($\sigma = 0.1$). When discarding the 20% of the discordant human subjects in both questionnaires ($\sigma = 0.2$), the system's polarities are very concordant with the average polarity reported by the subject (precision > 0.85). With $\sigma \geq 0.4$, the resulting precision is 0.93 for both questionnaires. Such high precision supports the ability of the approach to detect positive or negative opinions in the reviews about specific features. As the polarity precision obtained in the two questionnaires is very similar (see Figure 7), the performance of the proposed approach is consistent across different reviews and features.

Taking into account the disagreement in the responses with the threshold σ, the aforementioned precision and recall values prove the efficacy of the proposed approach. This confirms that the approach is capable of recognizing opinionated features, and of correctly identifying the polarity sign. The next section discusses the ability of the system to detect the correct degree of polarity expressed about a feature.

	Q A	Q B
Number of human subjects N	21	18
Human min $sim(P_h, \bar{P}_h)$	.66	.68
Human mean $sim(P_h, \bar{P}_h)$	.89	.84
Human median $sim(P_h, \bar{P}_h)$	.89	.86
Human max $sim(P_h, \bar{P}_h)$	.97	.96
Machine $sim(P_m, \bar{P}_h)$	.82	.82

Table 5 Similarity between human (h) and machine (m) polarities

4.5 Polarity degree precision

The human subjects were asked to rate the author's sentiment on a five-point Likert scale, with two positive cases, two negative cases, and "not available." Hence, it is important to evaluate the system's ability to capture not only the polarity sign (positive, negative, or not available), but also the *polarity degree precision*, i.e. the intensity of the writer's sentiment towards a feature.

To assess the system's ability to quantify feature polarity degree precision, we use the cosine similarity measure between the human polarities P_h and the machine polarities P_m as an indicator of performance. First, we scale the human polarities P_h in the interval $[-1, 1]$. Second, we define the cosine measure $sim(V_1, V_2) \in [-1, 1]$, where V_1 and V_2 are vectors of polarities, and -1 indicates perfect inverse polarities, 0 no similarity, and 1 identical polarities.

We consider the mean human polarities $\bar{P}_h$ as the ground truth against which individual humans and the machine can be evaluated. The similarity $sim(P_h, \bar{P}_h)$ indicates the performance of an individual human, whilst $sim(P_m, \bar{P}_h)$ indicates the machine performance. To observe the machine performance in context, we compute all the human and machine similarities in groups A and B. The similarity scores of machine and humans are summarized in Table 5.

Restricting the analysis to the human performance, it is possible to notice that the human subjects obtained a $sim(P_h, \bar{P}_h) \in [0.66, 0.97]$. The vast majority of human subjects performed the task similarly to the average ($sim \approx .86$), with a tail of exceptionally good subjects ($sim > 0.9$), and a tail of exceptionally bad subjects ($sim < 0.7$). The distribution of these human similarities can be analyzed via a kernel density estimate (KDE), using a univariate Gaussian smoothing kernel. The resulting density function captures the distribution of the similarities, highlighting the general trends present in the data. The KDE of the similarities is depicted in Figure 8, showing the similarity distribution for both questionnaires A and B.

The ability of the system to capture the polarity degree can be observed in the human performance as the context. The similarity between the polarities returned by the system P_m and the human mean $\bar{P}_h$ is ≈ 0.82 both for questionnaire A and B. This result is represented in Figure 8 as a vertical line. Whilst these similarity scores are slightly lower than the human average (respectively 0.89 and 0.84), the system performs comparably to the human subjects in both questionnaires. Taking into account the variability of polarity degree in the human responses, this result strongly indicates that the proposed approach to computing polarities at the feature level can be compared favorably to human performance.

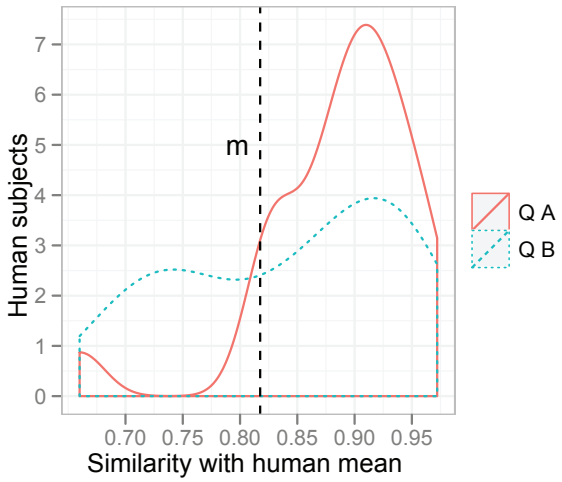

Fig. 8 Kernel density estimate (KDE) of the similarity of the human subjects against the human mean, for questionnaire A ('Q A') and B ('Q B'). The vertical line indicates the machine performance ('m').

5 Conclusions

In this paper, we presented a novel approach to detecting polarities in reviews at the feature level. The approach integrates natural language processing techniques with statistical approaches, in order to recognize opinions, and compute polarities about specific aspects of user-generated reviews (see Section 3). First, we presented a method for the automatic extraction of features from a domain-specific corpus. This extraction method enables the detection of the most salient features of a product or a service from the user's perspective. Subsequently, for each review, our sentiment detection approach is able to identify which features have been commented on ('opinionated'), and can estimate their local polarity via related terms that express a non-neutral opinion. To achieve this, we modeled each

sentence as a set of terms in a dependency graph, connected through syntactic and semantic dependency relations.

To evaluate our approach, we carried out a user study on a corpus of user-generated reviews, comparing the system's results against 39 human subjects. We showed the validity and reliability of the proposed method based on the system's polarity precision, recall, and degree, observing the performance of the system from several viewpoints, and we have released the human-generated dataset online.[8] In this evaluation, the approach obtains high precision and recall on the features, and computed the polarity degree only slightly below the average human performance.

Although the evaluation focused on hotel reviews, our approach could be used to extract sentiment patterns from social networks. Insights about the users' opinions can be gained from informal reviews generated in fast streams on Twitter, Weibo, and Facebook, and in slower streams of online reviews, such as Epinions and RateItAll. As discussed in Section 2, our approach can be used to study how opinions spread in dynamic, complex, open social systems. Unlike often uninterpretable overall document polarities, the detection of fine-grained features provides a sophisticated analytic instrument, which may benefit both consumers and service providers.

Considering future directions for this work, challenging problems and limitations of the approach need to be addressed. The high linguistic and cultural variability of texts generated in online social networks poses considerable issues for automated sentiment analysis techniques. Notably, users often misspell words, use idiosyncratic spelling variants, and avoid standard grammatical structures, resulting in uninterpretable dependency graphs. Moreover, opinions can be presented through irony and sarcasm, causing mistakes in the sentiment analysis (Carvalho et al. 2009). In international open platforms, users express themselves in a variety of native and non-native languages, resulting in complex, non-standard mixtures of linguistic styles that defy traditional natural language processing tools (Warschauer et al. 2010).

Another major issue is the context-dependency of term polarities. Using WordNet, our system takes into account the different word senses, distinguishing for example between 'bar' as the noun referring to an establishment where alcoholic drinks are served, and 'bar' as the verb meaning 'prevent from entering.' However, in many situations polarities cannot be assigned to the word senses in isolation, without considering a domain and a linguistic context. For example, while 'good' conveys positive polarity in most cases, the word 'fast' can indicate a positive ("The service is fast and efficient") or a negative opinion ("the battery of my camera runs out fast"). More advanced, context-dependent linguistic models of polarity are needed (see, for example, Zhang and Liu 2011).

Finally, our approach cannot distinguish between authentic and fake reviews, e.g. positive reviews written by service providers or spam. The unconstrained proliferation of fake online reviews is a major issue both for consumers and social networks' administrators, and constitutes a relevant research challenge that has yet not found satisfactory solutions (Mukherjee et al. 2012). In order to cope with this important issue, the sentiment detection process should be coupled with models of the user's reputation. The quantification of the user reputation in a weight can provide a leverage to refine the sentiment analysis towards the extraction of more truthful human opinions.

[8] `http://github.com/ucd-spatial/Datasets`

References

Annett, M., Kondrak, G. (2008). A comparison of sentiment analysis techniques: Polarizing movie blogs. In: *Advances in Artificial Intelligence* (pp. 25–35), Springer, LNCS, vol. 5032.

Baccianella, S., Esuli, A., Sebastiani, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)* (pp. 2200–2204).

Baldwin, T., Lui, M. (2010). Language Identification: The Long and the Short of the Matter. In: *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 229–237), ACL.

Banerjee, M., Capozzoli, M., McSweeney, L., Sinha, D. (1999). Beyond kappa: A review of interrater agreement measures. *Canadian Journal of Statistics*, *27*(1), 3–23.

Beineke, P., Hastie, T., Manning, C., Vaithyanathan, S. (2004). Exploring Sentiment Summarization. In: *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications* (pp. 1–4), AAAI.

Carvalho, P., Sarmento, L., Silva, M., de Oliveira, E. (2009). Clues for Detecting Irony in User-Generated Contents: Oh...!! It's so easy ;-). In: *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion* (pp. 53–56), ACM.

Chen, L., Qi, L. (2011). Social Opinion Mining for Supporting Buyers' Complex Decision Making: Exploratory User Study and Algorithm Comparison. *Social Network Analysis and Mining*, *1*(4), 301–320.

Chevalier, J., Mayzlin, D. (2006). The Effect of Word of Mouth on Sales: Online Book Reviews. *Journal of Marketing Research*, *43*(3), 345–354.

Collins, M. J. (1999). Head-driven statistical models for natural language parsing. PhD thesis, University of Pennsylvania, Philadelphia, PA, USA.

Dawes, J. (2008). Do Data Characteristics Change According to the Number of Scale Points Used? An Experiment Using 5 Point, 7 Point and 10 Point Scales. *International Journal of Market Research*, *51*(1).

De Marneffe, M. C., Maccartney, B., Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In: *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC 2006)* (pp. 449–454).

Ding, X., Liu, B. (2010). Resolving object and attribute coreference in opinion mining. In: *Proceedings of the 23rd International Conference on Computational Linguistics* (pp. 268–276), ACL.

Fellbaum, C. (ed.) (1998). *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.

Fleiss, J. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, *76*(5), 378–382.

Ganesan, K., Zhai, C. (2012). Opinion-based entity ranking. *Information retrieval*, *15*(2), 116–150.

Godbole, N., Srinivasaiah, M., Skiena, S. (2007). Large-scale sentiment analysis for news and blogs. In: *Proceedings of the International Conference on Weblogs and Social Media (ICWSM)* (pp. 219–222).

Hatzivassiloglou, V., McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In: *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics* (pp. 174–181), ACL, EACL '97.

Hiroshi, K., Tetsuya, N., Hideo, W. (2004). Deeper Sentiment Analysis Using Machine Translation Technology. In: *Proceedings of the 20th International Conference on Computational Linguistics* (pp. 1–7), ACL, COLING '04.

Holz, F., Teresniak, S. (2010). Towards Automatic Detection and Tracking of Topic Change. In: *Computational Linguistics and Intelligent Text Processing* (pp. 327–339), Springer, LNCS, vol. 6008.

Hu, M., Liu, B. (2004a). Mining and summarizing customer reviews. In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 168–177), ACM, KDD '04.

Hu, M., Liu, B. (2004b). Mining opinion features in customer reviews. In: *Proceedings of the 19th national conference on Artifical intelligence (AAAI'04)* (pp. 755–760), AAAI.

Kannan, K., Goyal, M., Jacob, G. (2012). Modeling the impact of review dynamics on utility value of a product. *Social Network Analysis and Mining*, pp. 1–18.

Kasami, T. (1965). An efficient recognition and syntax analysis algorithm for context-free languages. Technical Report AFCRL-65-758, Air Force Cambridge Research Laboratory.

Klein, D., Manning, C. D. (2003). Accurate unlexicalized parsing. In: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics* (pp. 423–430), ACL, ACL '03.

Lipsman, A. (2007). Online Consumer-Generated Reviews Have Significant Impact on Offline Purchase Behavior (comScore, Inc. and The Kelsey Group). URL `http://www.comscore.com/Insights/Press_Releases/2007/11/Online_Consumer_Reviews_Impact_Offline_Purchasing_Behavior`.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167.

Liu, B., Hu, M., Cheng, J. (2005). Opinion Observer: Analyzing and Comparing Opinions on the Web. In: *Proceedings of the 14th International Conference on World Wide Web* (pp. 342–351), ACM, WWW '05.

Marcus, M. P., Marcinkiewicz, M. A., Santorini, B. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, *19*(2), 313–330.

Matsumoto, S., Takamura, H., Okumura, M. (2005). Sentiment Classification Using Word Sub-sequences and Dependency Sub-trees. In: *Proceedings of the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining* (pp. 301–311), Springer, PAKDD'05.

McDonald, R., Nivre, J. (2011). Analyzing and integrating dependency parsers. *Comput Linguist*, *37*(1), 197–230.

Miao, Q., Li, Q., Dai, R. (2009). AMAZING: A sentiment mining and retrieval system. *Expert Systems with Applications*, *36*(3), 7192–7198.

Missen, M., Boughanem, M., Cabanac, G. (2012). Opinion Mining: Reviewed from Word to Document Level. *Social Network Analysis and Mining*, pp. 1–19.

Moilanen, K., Pulman, S. (2007). Sentiment Composition. In: *Proceedings of the Recent Advances in Natural Language Processing International Conference (RANLP 2007)* (pp. 378–382).

Morinaga, S., Yamanishi, K., Tateishi, K., Fukushima, T. (2002). Mining product reputations on the Web. In: *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 341–349), ACM, KDD '02.

Mukherjee, A., Liu, B., Glance, N. (2012). Spotting Fake reviewer groups in consumer reviews. In: *Proceedings of the 21st international conference on World Wide Web (WWW 2012)* (pp. 191–200), ACM.

Nadeau, D., Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, *30*(1), 3–26.

O'Connor, P. (2010). Managing a hotel's image on tripadvisor. *Journal of Hospitality Marketing & Management*, *19*(7), 754–772.

Oelke, D., Hao, M. C., Rohrdantz, C., Keim, D. A., Dayal, U., Haug, L. E., Janetzko, H. (2009). Visual opinion analysis of customer feedback data. In: *IEEE VAST* (pp. 187–194).

Pang, B., Lee, L. (2004). A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics* (pp. 271–278), ACL, ACL '04.

Pang, B., Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, *2*(1-2), 1–135.

Pang, B., Lee, L., Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In: *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing* (pp. 79–86), ACL, EMNLP '02, vol. 10.

Pedersen, T., Kolhatkar, V. (2009). WordNet::SenseRelate::AllWords: a broad coverage word sense tagger that maximizes semantic relatedness. In: *The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 17–20), ACL.

Pekar, V., Ou, S. (2008). Discovery of subjective evaluations of product features in hotel reviews. *Journal of Vacation Marketing*, *14*(2), 145–155.

Ponzetto, S. P., Strube, M. (2007). An API for measuring the relatedness of words in Wikipedia. In: *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics on Interactive Poster and Demonstra-*

tion Sessions (pp. 49–52), ACL.
Popescu, A. M., Etzioni, O. (2005). Extracting Product Features and Opinions from Reviews. In: *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing* (pp. 339–346), ACL, HLT '05.
Qiu, G., Liu, B., Bu, J., Chen, C. (2009). Expanding Domain Sentiment Lexicon Through Double Propagation. In: *Proceedings of the 21st International Joint Conference on Artifical Intelligence* (pp. 1199–1204), Morgan Kaufmann Publishers Inc..
Qiu, G., Liu, B., Bu, J., Chen, C. (2011). Opinion Word Expansion and Target Extraction through Double Propagation. *Computational Linguistics*, *37*(1), 9–27.
Salton, G., Buckley, C. (1988). Term-weighting Approaches in Automatic Text Retrieval. In: *Information Processing and Management* (pp. 513–523), vol. 24.
Titov, I., McDonald, R. (2008). Modeling online reviews with multi-grain topic models. In: *Proceedings of the 17th international conference on World Wide Web* (pp. 111–120), ACM.
Turney, P. D. (2002). Thumbs Up or Thumbs Down?: Semantic Orientation Applied to Unsupervised Classification of Reviews. In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 417–424), ACL.
Turney, P. D., Littman, M. L. (2003). Measuring Praise and Criticism: Inference of Semantic Orientation from Association. *ACM Trans Inf Syst*, *21*(4), 315–346.
Warschauer, M., Black, R., Chou, Y. (2010). Online Englishes. In: Kirkpatrick, T. (ed.) *The Routledge Handbook of World Englishes*, New York: Routledge, pp. 490–505.
Wu, Y., Wei, F., Liu, S., Au, N., Cui, W., Zhou, H., Qu, H. (2010). OpinionSeer: Interactive Visualization of Hotel Customer Feedback. *IEEE Transactions on Visualization and Computer Graphics*, *16*(6), 1109–1118.
Ye, Q., Law, R., Gu, B. (2009). The impact of online user reviews on hotel room sales. *International Journal of Hospitality Management*, *28*(1), 180–182.
Zhai, Z., Liu, B., Xu, H., Jia, P. (2011a). Clustering product features for opinion mining. In: *Proceedings of the 4th ACM International Conference on Web Search and Data Mining* (pp. 347–354), ACM.
Zhai, Z., Liu, B., Xu, H., Jia, P. (2011b). Constrained LDA for Grouping Product Features in Opinion Mining. *Advances in Knowledge Discovery and Data Mining*, pp. 448–459.
Zhang, L., Liu, B. (2011). Identifying noun product features that imply opinions. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers* (pp. 575–580), vol. 2.
Zhang, L., Liu, B., Lim, S., O'Brien-Strain, E. (2010). Extracting and ranking product features in opinion documents. In: *Proceedings of the 23rd International Conference on Computational Linguistics: Posters* (pp. 1462–1470), ACL.
Zhou, L., Chaovalit, P. (2008). Ontology-supported polarity mining. *Journal of the American Society for Information Science and Technology*, *59*(1), 98–110.