A Language Independent Method for Generating Large Scale Polarity Lexicons

Giuseppe Castellucci, Danilo Croce, Roberto Basili
Dept. of Electronic Engineering, Dept. of Enterprise Engineering, Dept. of Enterprise Engineering
University of Roma, Tor Vergata, Via del Politecnico 1, 00133 Roma, Italy
castellucci@ing.uniroma2.it, croce@info.uniroma2.it, basili@info.uniroma2.it

Abstract

Sentiment Analysis systems aims at detecting opinions and sentiments that are expressed in texts. Many approaches in literature are based on resources that model the *prior* polarity of words or multi-word expressions, i.e. a polarity lexicon. Such resources are defined by teams of annotators, i.e. a manual annotation is provided to associate emotional or sentiment facets to the lexicon entries. The development of such lexicons is an expensive and language dependent process, making their coverage of linguistic sentiment phenomena limited. Moreover, once a lexicon is defined it can hardly be adopted in a different language or even a different domain. In this paper, we present several Distributional Polarity Lexicons (DPLs), i.e. large-scale polarity lexicons acquired with an unsupervised methodology based on Distributional Models of Lexical Semantics. Given a set of heuristically annotated sentences from Twitter, we transfer the sentiment information from sentences to words. The approach is mostly unsupervised, and experimental evaluations on Sentiment Analysis tasks in two languages show the benefits of the generated resources. The generated DPLs are publicly available in English and Italian.

Keywords: Polarity Lexicon Generation, Sentiment Analysis, Distributional Models

1. Introduction

Sentiment Analysis (SA) and Opinion Mining systems (Pang and Lee, 2008) aim at tracking the opinions expressed in texts with respect to specific topics, e.g. products or people. In particular, SA deals with the problem of deciding whether a portion of a text, e.g. a sentence or a phrase, is expressing a trend towards specific feelings, e.g. *positivity* or *negativity*. Polarity lexicons are list of words each associated to one or more values, indicating their trend towards specific feelings, e.g. a single value in $[-1, +1]$ can be used to indicate *negativity* (-1), *neutrality* (0) or positivity $(+1)$. Such lexicons have been defined to support the development of automatic systems for detecting subjective phrases or sentences and recognizing their polarity (Mohammad and Turney, 2010). For example, in these lexicons, "*good*" can be associated to a prior positive sentiment in contrast to "*sad*", considered negative in every domain. The occurrence of such words in a sentence can be adopted as an indicator of the polarity trends that are expressed in the sentence. These lexicons are often hand-compiled; however, from a linguistic point of view, a-priori membership of words to polarity classes can be considered too restrictive, as sentiment expressions are often topic dependent: the occurrences of the word *mouse*, for example, are mostly neutral in the consumer electronics domain, while it can be negatively biased in a restaurant domain. Representing topic specific polarity phenomena would require manual revisions of the lexicon entries, thus resulting in a costly operation. Moreover, these resources mainly exist for the English language, while they scarce for the others. Again, the manual annotation of a polarity lexicon for a new language can be an expensive process that cannot be afforded in many cases.

In this paper, we describe large-scale polarity lexicons resources that have been released to the research community. They have been acquired with a methodology we recently proposed (Castellucci et al., 2015). It consists of an unsupervised methodology to derive large-scale polarity lexicons, which exploits the extra-linguistic information within Social Media data, e.g. the presence of emoticons in messages. The approach relies on Distributional Models of Lexical Semantics (Sahlgren, 2006; Mikolov et al., 2013b), where the equivalence in sentences and words representations available in some distributional models (e.g. the dual LSA space for words and texts introduced in (Landauer and Dumais, 1997)) is exploited to transfer sentiment information from sentences to words. In fact, as sentences can be clearly related to polarity, a classifier can always be trained in such spaces and used to transfer sentiment information from sentences to words. Specifically, we train a polarity classifier by observing sentences expressing some polarity and we adopt it to classify words in order to populate a polarity lexicon. Annotated messages are derived from Twitter[1] and their polarity is determined by simple heuristics. It means that words in specific domains can be related to sentiment classes by looking at their semantic closeness to emotionally biased sentences. The approach for deriving a new lexicon is highly applicable, language and domain independent, as the distributional model can be acquired without any supervision, and the provided heuristics do not have any bias with respect to languages or domains. Thus, polarity lexicons can be generated in multiple languages, without the need of expensive manual supervision.

We generated large-scale polarity lexicons for English and Italian and we released them to the research community[2]. Their contribution is measured against different SA tasks in the two languages. In particular, our evaluation is based on Twitter Sentiment Analysis, as recently it has been the focus of highly participated challenges, as the recent SemEval (Nakov et al., 2013; Rosenthal et al., 2014) and Evalita (Basile et al., 2014) tasks demonstrate.

[1] http://www.twitter.com

[2] http://sag.art.uniroma2.it/demo-software/distributional-polarity-lexicon/

In the rest of the paper, related works are discussed in Section 2., while Section 3. presents the methodology proposed to generate the lexicons. In Section 4. a detailed description of the generated lexicons is provided. The beneficial impact of these resources in polarity detection tasks is discussed in Section 5.. Finally, in Section 6., conclusions and future works are presented.

2. Related Works

Polarity lexicon generation has been tackled in many researches and three main approaches can be pointed out for producing a polarity lexicon.

Manually annotated lexicons. Earlier works in lexicon generation are based on manual annotations of terms with respect to emotional or sentiment categories. For example, in (Stone et al., 1966) sentiment labels are manually associated to 3600 English terms. In (Hu and Liu, 2004) a list of positive and negative words are manually extracted from customer reviews. The MPQA Subjectivity Lexicon (Wilson et al., 2005) contains words, each with its prior polarity (positive or negative) and a discrete strength (strong or weak) value. The NRC Emotion Lexicon (Mohammad and Turney, 2010) is composed by frequent English nouns, verbs, adjectives, and adverbs annotated through Amazon Mechanical Turk system with respect to eight emotions (e.g. joy, sadness, trust) and sentiment.

Lexicons acquired over graphs. Graph based approaches exploit an underlying semantic structure that can be built upon words. In (Esuli and Sebastiani, 2006) the WordNet (Miller, 1995) synset glosses are exploited to derive three scores describing the positivity, negativity and neutrality of the synsets. The work in (Rao and Ravichandran, 2009) generates a lexicon as graph label propagation problem. Each node in the graph represents a word. Each weighted edge encodes a relation between words derived from WordNet (Miller, 1995). The graph is constructed starting from a set of manually defined seeds. The polarity for the other words is determined by exploiting graph-based methods.

Corpus-based lexicons. Statistics based approaches are more general, as they mainly exploit corpus processing techniques. (Turney and Littman, 2003) proposed a minimally supervised approach to associate a polarity tendency to a word by determining if it co-occurs more with positive words than negative ones. More recently, (Zhang and Singh, 2014) proposed a semi-supervised framework for generating a domain-specific sentiment lexicon. Their system is initialized with a small set of labeled reviews, from which segments whose polarity is known are extracted. It exploits the relationships between consecutive segments to automatically generate a domain-specific sentiment lexicon. In (Kiritchenko et al., 2014) a minimally-supervised approach based on Social Media data is proposed by exploiting hashtags or emoticons related to positivity and negativity, e.g., `#happy`, `#sad`, `:)` or `:(`. They compute a score, reflecting the polarity of each word, through a Point wise Mutual Information based measure between a word and an emotion. In (Saif et al., 2014) word contexts are adopted to generate sentiment orientation for words. In particular, the sentiment of context words, available in an already built lexicon, is shown to contribute in deriving the sentiment orientation of a target word. As a result, the so-called *SentiCircle* is derived for each target word by considering the contexts in which they appear. The approach here presented can be seen as more general, as it does not rely on any existing lexicon, but it could be used to build a *SentiCircle*.

3. Polarity Lexicons Generation

In this Section, the approach defined in (Castellucci et al., 2015) for the Automatic Generation of Polarity Lexicons is described. The generation process relies on an *embedded* representation of lexical items, that is derived by approaches in the family of the Distributional Model (DM) of Lexical Semantics; in these models, semantics relationships between words and sentences can be exploited at the same time. In particular, the approach is based on the possibility to acquire comparable representations (e.g. vectors) for sentences and words. In this perspective, common algebraic operations are then adopted to establish relationships between lexical items and polarity biased sentences.

3.1. Word Embedding for Lexical Semantics

Word embedding are adopted to represent lexical items in compact vector representations. These vectors *embed*, within the dimensions of the space, information about the semantic relationships of words; this information is acquired by looking at the words usage in large corpora. Word embedding fall in the family of the Distributional Models (DMs) of Lexical Semantics: the foundation for these models is the *Distributional Hypothesis* (Harris, 1964), i.e. words that are used and occur in the same contexts tend to have similar meanings. Although different DMs have been defined, they are similar in nature as they all derive vector representations for words, i.e. a word space, from more or less complex corpus processing stages.

The semantic relationships of interest are derived in terms of algebraic operations in the word space, such as the cosine similarity. Different relationships can be modeled depending on the way the word space has been acquired. For example, *topical* similarities between words can be captured when vectors are built considering the occurrence of a word in documents; *paradigmatic* similarities can be, instead, captured when vectors are built considering the occurrence of a word in the context of another word (Sahlgren, 2006). In such models, words like *run* and *walk* are close in the word space, while *run* and *read* are projected in different sub-spaces. These representations can be derived mainly in two ways[3]: *counting* the co-occurrences between words, e.g. (Landauer and Dumais, 1997; Sahlgren, 2006), and then, optionally, applying dimensionality reduction techniques, such as Singular Value Decomposition (Golub and Kahan, 1965) in Latent Semantic Analysis (Landauer and Dumais, 1997). Another popular method for the acquisition of word spaces relies on a supervised setting, where the *prediction* of context-based task is exploited (examples are (Bengio et al., 2003; Mikolov et al., 2013a)). These are intended to capture more *syntagmatic* aspects during the word space construction.

[3] For an in-depth comparison between the two methods, refer to (Baroni et al., 2014)

Despite the specific algorithm used for the space acquisition[4], all these approaches allow to derive a projection function $\Phi(\cdot)$ of words into a geometrical space, i.e. the d-dimensional vector representation for a word $w_k \in \mathbb{W}$ is obtained through $\vec{w_k} = \Phi(w_k)$. Geometrical regularities are exploited to determine the prior sentiment for words, i.e. the main assumption is that words carrying sentiment lie in specific sub-spaces. In the following, we discuss how polarity transfer can be applied from sentences (whose polarity is known) to single words by exploiting those sub-spaces. Polarized words often share the same contexts: as a consequence, a DM tends to derive similar representations for words characterized by opposite polarity, e.g. *joy* and *sorrow* (see the first and second columns in Table 6). The final aim is thus to leverage on DMs, because of their ability to represent the semantic relationships between words, but considering also their sentiment in order to obtain a new representation that will account for sentiment-related phenomena.

3.2. Lexicon Generation through Classification

The semantic similarity (closeness) established by traditional DMs does not correspond well with emotional similarity. For example, in the Table 6 we selected words that are polarity carrier in English. In the second column we selected the three most similar words to each of them according to the cosine similarity measure computed between vectors derived from the DM. As it can be noticed, each word is similar to another word whose polarity can be defined as *opposite* to the one of the target word. Instead, sentiment or emotional differences between words should be captured into representations that are able to coherently express the underlying sentiment. In this perspective, a discriminant function can be derived by machine learning over these representations. Let us consider a space $\mathbb{R}^d$ where some geometrical representation of a set of annotated examples can be derived. In general, a linear classifier can be seen as a separating hyper plane $\theta \in \mathbb{R}^d$ that is used to classify a new example represented in the same space. Each θ_i corresponds to a specific dimension, or feature i that has been extracted from the annotated examples. After a learning stage, the magnitude of each θ_i reflects the importance of the feature i with respect to a target phenomenon. In this sense, when applied on distributional vectors of word semantics, linear classifiers are expected to learn such regions that are useful to properly discriminate examples with respect to the target classes. If these classes reflect the sentiment expressed by words, a classifier should find those sub-spaces better correlating examples with the sentiment. In this way, any set of words w_i in the vocabulary $\mathbb{W}$ associated with their prior polarity could be used to train a sentiment classifier. In fact, given a set of seed words whose prior polarity is known, their projection in the Word Space model $\vec{w_k}^{seed} = \Phi(w_k^{seed})$ is sufficient to train the linear classifier. This would find what dimensions in $\mathbb{R}^d$ are related to the different polarities. Classification thus corresponds to transferring the knowledge about sentiment implicit in the seed words to the other remaining words. The definition and annotation of seed words, however, could be expensive and hardly portable across natural languages. Moreover, lexical items do change emotional flavor across domains and the knowledge embodied by the seed lexicons may not generalize when different domains are faced. We suggest to avoid the selection of lexical seeds and emphasize the role of distributional models: a common representation of sentences and words is here capitalized to automatize the development of portable sentiment lexicons. In (Castellucci et al., 2015) we propose to make use of sentences as the training material of the classifier, as they embody sentiment more explicitly: for example, sentences including strong sentiment markers can be cheaply gathered, thus providing a large scale seed resource. As these sentences and words (i.e. candidate entries for the polarity lexicon) lie into the same space (i.e. sentences and semantically related words belong to the same sub-spaces), we train a classifier over sentences and apply it to produce a very large lexicon. Sub-spaces strongly related to a sentiment class are captured from the analysis of the sentences, and they are used to project the sentiment over the lexicon. In details, we have words w_k in the vocabulary $\mathbb{W}$ and their vector representation $\vec{w_k} \in \mathbb{R}^d$ obtained by projecting them in a Word Space, i.e. $\vec{w_k} = \Phi(w_k)$. We also have a training set $\mathbb{T}$, including sentences associated to a polarity class. In order to project an entire sentence in the same space, we apply a simple but effective linear combination operator. For each sentence $t \in \mathbb{T}$, we derive the vector representation $\vec{t} \in \mathbb{R}^d$ by summing all the word vectors composing the sentence, i.e. $\vec{t} = \sum_{w_i \in t} \Phi(w_i)$. It is one of the simpler, but still expressive, method that is used to derive a representation that accounts for the underlying meaning of a sentence, as discussed in (Landauer and Dumais, 1997). Having projected an entire sentence in the space, we can find all the dimensions of the space that are related to a sentiment class. Sentence representations are adopted to train a linear discriminant function f expected to capture the sentiment related sub-spaces by properly weighting each dimension i of the original space. The lexicon is generated by applying f to the entire $\mathbb{W}$. As we deal with multiple sentiment classes, f can be seen as m distinct binary classification functions $(f_1, \ldots, f_m)$, one for each sentiment class. Each word $w \in \mathbb{W}$ is classified with all the f_i, thus receiving m distinct scores $f_i(w)$ reflecting the classifier confidence in the membership of w to class i. The final normalized polarity score $o_i(w)$ is obtained from $f_i(w)$ through a softmax function[5]: each w can be represented both with its distributional representation, i.e. $\vec{w} = \Phi(w)$, and its sentiment representation, i.e. $\vec{o}(w)$.

Generating a Dataset through Emoticons An annotated dataset of sentences $\mathbb{T}$ is needed to acquire a linear classifier that emphasizes specific sub-spaces. Although different datasets of such kind exist, our aim is to use a general methodology that can enable the use of this technique in different domains or languages. We are going to use heuristic rules to select sentences by exploring Twitter messages through the emoticons, i.e. a Distant Supervision approach (Go et al., 2009). In order to derive messages belonging to

[4] In this paper we will acquire word spaces with the *prediction* based technique.

[5] $o_i(w) = e^{f_i(w)} / \sum_{j=1}^{m} e^{f_j(w)}$

the positive or negative classes, we select Twitter messages whose last token is a smile either positive, e.g. `:)` or `:D` or negative, e.g. `:(` or `:-(`. Neutral messages are filtered by looking at those messages that end with a url, as in many cases these are written by newspaper accounts and they use mainly non-polar words to announce an article. In order to have a more accurate dataset, we filter out those messages that contain elements of other classes, i.e. if a message ends with a positive smile and it contains either a negative smile or a url it will be discarded.

4. Distributional Polarity Lexicons for English and Italian

The methodology presented in the Section 3., has been applied to derive polarity lexicons in two languages, i.e. English and Italian. For each language, we release two configurations of the lexicon: the first version is composed by words that are lemmatized and pos-tagged. We will call these pre-processed lexicons `DPLp-EN` and `DPLp-IT`, respectively for English and Italian. A second version contains words that have been not pre-processed. We will refer to this lexicon as `DPL-EN` and `DPL-IT`, respectively for English and Italian.

All the lexicons are derived from the analysis of corpora of Twitter messages[6] that have been gathered through the Twitter API. We gathered about 30 million of messages in English and 10 million in Italian. In order to acquire the lemmatized and pos-tagged versions of the lexicons, each corpus has been pre-processed with a multi-lingual parser, i.e. the Chaos system (Basili and Zanzotto, 2002), that has been adapted to manage Twitter specific phenomena, e.g. hashtags or user mentions. The corpora have been also filtered out with the heuristics based on the presence of emoticons and links described in the previous section. From each corpus, we derived $10,000$ messages for each *sentiment* class, i.e. *positive*, *negative* and *neutral* that have been adopted to train the linear classifier f.

Word spaces are derived according to a specific Distributional Model, that is the Skip-gram model defined in (Mikolov et al., 2013a) with the `word2vec` tool[7]. In particular, we derived 250-dimensional word vectors, by analyzing these corpora made of tweets. For each language, we derived two word spaces from each corpus, i.e. one with words with their original surface and one with the lemmatized and pos-tagged words.

Lexicons are generated by applying the methodology described in the Section 3.: for each space, a linear classifier f is first acquired through a linear Support Vector Machine[8] applied on vectors representing the set of tweets selected via heuristics; then, each word represented in the same word space is classified through f to derive its polarity scores.

In Table 1, statistics related to the English and Italian lexicons independent from pos-tags and lemmatizations are shown (`DPL-EN` and `DPL-IT`). The English word space is composed by $191,389$ words, each of which has been classified by the f classifier to produce the corresponding lexicon. In Italian, the word space, and consequently the polarity lexicon, is composed by $143,764$ words. In this Table, statistics about the distribution with respect to the three polarity classes *positive*, *negative* and *neutral* are shown. We decided a word belongs to a specific polarity class with the argmax operator, i.e. $pol(w) = \arg\max_c dpl\_c(w)$, where $c \in \{positive, negative, neutral\}$, and $dpl\_c(w)$ is the score of the word w with respect to c.

Language	Words	Positive	Negative	Neutral
`DPL-EN`	191,389	57,726	43,172	90,491
`DPL-IT`	143,764	34,790	36,188	72,786

Table 1: Statistics for the English and Italian lexicons when no pre-processing stages are applied.

Part-of-speech	Positive	Negative	Neutral
Verb	2,296	3,273	3,902
Noun	7,650	9,293	40,386
Adjectives	3,597	5,179	7,708
Adverbs	492	664	660
Hashtags	1,956	1,530	17,531

Table 2: Statistics for the English polarity lexicon (`DPLp-EN`) when the pre-processing stage is applied to derive lemmas and part-of-speech of each word.

The statistics of the `DPLp-EN` and `DPLp-IT` lexicons are instead shown in the Tables 2 and 3. The word spaces size was $188,635$ and $99,410$, respectively for English and Italian. We filtered out some words, maintaining only the *verbs*, *nouns*, *adjectives*, *adverbs* and *hashtags*. Thus, the final lexicon sizes are $106,117$ and $75,021$, respectively for English and Italian. The number of items in each class is reported in the Tables. Again, we decided the class membership of a word with the $\arg\max$ operator over the DPL scores.

Part-of-speech	Positive	Negative	Neutral
Verb	2,019	2,585	3,990
Noun	5,249	6,183	31,582
Adjectives	2,565	2,699	5,205
Adverbs	244	256	224
Hashtags	398	556	11,266

Table 3: Statistics for the Italian polarity lexicon (`DPLp-IT`) when the pre-processing stage is applied to derive lemmas and part-of-speech of each word.

In order to further demonstrate the effectiveness of the proposed resources, Table 4 shows some examples of words that can be found in the lexicon for the English language when the linguistic pre-processing is applied. We selected 6 words that are biased towards some sentiment in the `DPLp-EN` lexicon. In the Table 4 we report each word along with their polarity lexicon scores, that are derived after applying the softmax operator, as described in Section 3.. Notice, for example, that a highly polar word as the ad-

[6] We normalized each message to reduce noise: in particular, elongated words , e.g. *coool*, are normalized to their original form, e.g. *cool*.

[7] `https://code.google.com/archive/p/word2vec/`

[8] We adopted the linear SVM formulation available in KeLP (Filice et al., 2015)

Term	Positive	Negative	Neutral
good::j	0.74	0.11	0.15
:)	0.86	0.04	0.10
bad::j	0.12	0.80	0.08
pain::n	0.13	0.76	0.11
#apple::h	0.14	0.16	0.70
#ibm::h	0.07	0.04	0.89
#microsoft::h	0.09	0.09	0.82
#google::h	0.14	0.17	0.69
#dell::h	0.13	0.20	0.67
#barackobama::h	0.19	0.07	0.74
#mccain::h	0.22	0.16	0.62
article::n	0.16	0.09	0.75
government::n	0.09	0.09	0.82
friend::n	0.37	0.31	0.32
surprise::n	0.40	0.31	0.29

Table 4: Example of polarity lexicon terms and relative sentiment scores (English language) in `DPLp-EN`.

Term	Positive	Negative	Neutral
buono::j	0.77	0.12	0.11
:)	0.73	0.08	0.19
cattivo::j	0.23	0.63	0.14
sofferenza::n	0.17	0.48	0.35
#apple::h	0.17	0.12	0.71
#ibm::h	0.15	0.13	0.72
#microsoft::h	0.14	0.12	0.74
#google::h	0.20	0.07	0.73
#dell::h	0.24	0.09	0.67
#barackobama::h	0.09	0.07	0.84
#mccain::h	0.13	0.02	0.85
articolo::n	0.19	0.05	0.76
governo::n	0.12	0.12	0.76
amico::n	0.44	0.24	0.32
sorpresa::n	0.40	0.22	0.38

Table 5: Example of polarity lexicon terms and relative sentiment scores (Italian language) in `DPLp-IT`.

jective *good* (in the first row) has a bias towards positivity, as demonstrated by the *Positive* score of 0.74 (in the second column), while it is less biased towards the *Negative* class, whose score is 0.11. Similar trends can be pointed out for other words, such as the noun *pain* (fourth row in the table), which is more biased towards negativity, as demonstrated by the *Negative* score of 0.76. These polarity scores are reasonable if compared to a human assignment of the polarity to the same words. For example, let us consider the Subjectivity Lexicon (Wilson et al., 2005), where a word is manually associated to a polarity category (positive, negative, neutral) with a strength value (weak or strong). The polarity values in this lexicon for *good* and *pain* are *weak positive* and *strong negative*, respectively. The `DPLp-EN` polarity assignment for these two words is comparable to the ones in the Subjectivity lexicon.

Similar outcomes can be found in the Italian language lexicon, i.e. `DPLp-IT`. In Table 5 we report the polarity scores that can be found in the lexicon, for the translation in Italian of the words of Table 4. Again, the scores associated to each word are reasonable. For example, let us consider the adjective *cattivo* in the third row of the Table: it is clearly a word evoking negativity in the Italian language. The Distributional Polarity Lexicon approach is able to derive a *Negative* score of 0.63 (second column). Moreover, the English and Italian lexicons, i.e. `DPLp-EN` and `DPLp-IT`, assign similar scores to words that are the translation of each other. As an example, let us consider the adjective *good* in English and its translation in Italian (that is *buono*) in the first rows from the two Tables 4 and 5. The two lexicons assign to these words similar scores for all the three classes we are considering (0.74, 0.11, 0.15 and 0.77, 0.12, 0.11). Moreover, notice that companies related hashtags are all mainly neutral in both languages. The same phenomenon occurs with hashtags about politicians, as *#obama* or *#mccain*.

One of the aims of the Distributional Polarity Lexicon approach is to derive a new representation that is able of taking into account polarity phenomena of the lexical items. The specific scores assigned within the polarity lexicon, as the ones reported in Table 4 or 5, can be adopted as a new vector representing the sentiment of each word. This new representation can be used to enrich a pure semantically oriented representation of words (from the original DMs) in order to inject sentiment related information in the embedding. In other words, we can combine the original word space vectors and the DPL vectors in a way such that the resulting representation can still represent the semantics of the words being at the same time more suitable for sentiment analysis related tasks. In particular, for each word in the vocabulary, we derived such representation by juxtaposing[9] the original 250-dimensional word space vector, which is derived through `word2vec`, and a DPL vector: the result of this operation is a new 253-dimensional representation.

Term	w/o DPL	w/ DPL
joy (0.62,0.08,0.30)	happiness sorrow laughter	happiness positivity enjoyment
love (0.52,0.11,0.37)	adore luv hate	adore loves loove
worse (0.13,0.80,0.07)	better worser funnier	worser sadder shittier
sadly (0.07,0.91,0.02)	unfortunately alas thankfully	unfortunately alas nope

Table 6: Similar words in the embedding without ($2^{nd} column$) and with ($3^{rd} column$) DPL, whose scores (*positivity, negativity, neutrality*) are reported in parenthesis in the first column.

In Table 6 we computed the most similar words for some polarity-carrier word in English, or target word (first column in the Table). Given a vector representation of the target word, the K most similar items are the K nearest neighbors according to the cosine similarity measure. The second column refers to the most similar words of

[9] Each vector is normalized (i.e. converted to a unit vector) before the juxtaposition.

a target word when the corresponding vector representation is not enriched with the DPL, i.e. each word is represented through the original Skip-gram 250-dimensional vector. The third column of the Table instead reports the most similar words when the word space vector is enriched with the DPL-EN, i.e. when it is represented with the 253-dimensional vector that is derived after the operation of juxtaposition. Notice how the DPL enrichment makes opposite polarity words less similar in the vector space. For example, let us consider the target word *joy*, whose DPL scores are $(0.62, 0.08, 0.30)$, i.e. it is biased towards positivity. Notice, in the second column, that the original K-nearest neighbors contain the word *sorrow*, which can be considered a negative, and thus opposite, word. Thus, the original word space is not able to properly differentiate between these two words from a sentiment perspective. Notice, instead, that the enriched representation is more consistent with the underlying sentiment. In fact, the new 253-dimensional vector captures pure semantics phenomena, as the K-nearest neighbors are still related to the target, but it makes opposite polar words less similar in the space (see the third column in the Table). For example, *sorrow* is no more in the 3-nearest neighbors of *joy*, while its new 3 most similar words are *happiness*, *positivity* and *enjoyment*. It means that this new representation could be more suitable for representing words in Sentiment Analysis tasks.

5. Experimental Evaluation

In this Section, an indirect evaluation of the polarity lexicons is provided. In particular, all the released lexicons are used in Sentiment Analysis tasks in Twitter with respect to English and Italian. In each task, the recognition of sentiment classes underlying a tweet is modeled as a classification problem. As for the generation of Distributional Polarity Lexicons, the SVM learning algorithm implemented in KeLP (Filice et al., 2015) is adopted to acquire the classification functions.

5.1. Sentiment Analysis in Twitter in English

The DPL-EN and DPLp-EN lexicons are evaluated against two tasks, the Semeval 2013 and the Semeval 2014 Sentiment Analysis in Twitter tasks.

The evaluations are addressed with two basic feature representations in a SVM learning framework: a boolean Bag-of-Word (BoW) and a word space (WS) derived representation. The former captures pure lexical information, whereas each binary dimension expresses the presence (or absence) of a particular word in a sentence. The latter is based on a word space whose aim is to smooth the lexical overlap measure of the pure BoW. The WS representation of a sentence is obtained by summing the vectors of all its verbs, nouns, adjectives and adverbs. We further enrich these representations of a message with the information derived from the DPL. In the following evaluations we thus make two distinct representations that are dependent from the DPL-EN and DPLp-EN, each with a 3-dimensional feature vector obtained by summing the DPL values of all the words (for the case of DPL-EN) and of the verbs, nouns, adjectives and adverbs (for the case of DPLp-EN).

Kernel	**F1pn**	**F1pnn**
BoW	59.72	63.53
BoW+SBJ	61.46	64.95
BoW+DPL-EN	60.08	63.42
BoW+DPLp-EN	60.78	64.09
BoW+WS	66.12	68.56
BoW+WS+SBJ	65.20	67.93
BoW+WS+DPL-EN	65.90	68.38
BoW+WS+DPLp-EN	66.40	68.68
Best-System	69.02	-

Table 7: Twitter Sentiment Analysis 2013 results. *Best-System* refers to the top scoring system in SemEval 2013.

Kernel	**F1pn**	**F1pnn**
BoW	58.74	61.38
BoW+SBJ	60.82	62.85
BoW+DPL-EN	59.07	62.39
BoW+DPLp-EN	62.49	64.01
BoW+WS	65.20	66.35
BoW+WS+SBJ	64.29	66.13
BoW+WS+DPL-EN	64.76	67.18
BoW+WS+DPLp-EN	66.11	67.07
Best-System	70.96	-

Table 8: Twitter Sentiment Analysis 2014 results. *Best-System* refers to the top scoring system in SemEval 2014.

The SVM learning algorithm is adopted over a kernel function (Shawe-Taylor and Cristianini, 2004) that is the combination of linear kernels, each operating on a specific representation. As an example, the SVM may operate over a kernel that is the sum of the single linear kernels over the BoW, WS and DPLp-EN representations: we refer to such a setting as BoW+WS+DPLp-EN.

In Tables 7 and 8 the experimental outcomes for the 2013 and 2014 SemEval datasets are reported, as well as the Best-System in the two challenges. Performance measures are the *F1pn* and the *F1pnn*. The former is the arithmetic mean between the F1 measures of the positive and negative classes, i.e. the official score adopted by the SemEval challenges. The latter is the arithmetic mean between the F1 measures of the positive, negative and neutral classes. The WS representation is based on the same DM used to generate the polarity lexicon. Here, we compare the contribution of the DPLs with a well-known lexicon, i.e. the Subjectivity Lexicon by (Wilson et al., 2005). This resource is composed by words annotated with subjective polarity information (positive, negative, neutral) and a strength (weak or strong) value. To inject this information in the SVM learning algorithm, for each message we generate a new feature representation (SBJ) where each dimension refers to a polarity value with its relative strength. For example, the SBJ representation of "*Getting better!*" is a feature vector whose only dimension containing a value different from zero is associated to the category strong_pos.

In Table 7, results for the 2013 test dataset are shown, which is composed by $3,814$ examples. First, the baseline performance achievable with a linear kernel applied to the simple BoW representation is shown (63.53% *F1Pnn*). Then, the experimental results obtained by combining the other representations are reported. When applying the WS, an improvement can be noticed, as demonstrated by the *F1pnn* score of 68.56% in the BoW+WS kernel. It means that distributional representations are useful to capture the semantic phenomena behind sentiment-related expressions, even in short texts, and to alleviate data sparseness problems of the pure BoW kernel (as demonstrated by the ~ 5 points increment in *F1Pnn*). When combining also the DPLs, further improvements are obtained for both performance measures (66.40% *F1pn* and 68.68% in *F1pnn* when using the pre-processed lexicon). It seems that `DPLp-EN` effectively acts as a smoothing of the contribution of the pure lexical semantics information provided by WS. Notice that the adoption of the pre-processed version is more effective with respect to the adoption of the `DPL-EN`. The filtering applied to the part-of-speech helps in producing a better feature representation that is more suitable for the task of sentiment analysis, even with respect to noisy texts derived from Twitter. It is noticeable that the combination of BoW+WS+DPLp-EN would have produced a system ranking 2^{nd} in the 2013 SemEval challenge, where the *Best-system*[10] achieved the *F1pn* score of 69.02%.

Similar trends can be noticed for the 2014 test set, as shown in Table 8. In this case, we were not able to rely on the complete test set, as, at the time of this experimentation, some of the messages were no longer available for download. Our evaluation is carried out on $1,562$ test examples, while the full test set was composed by $1,853$. It makes a direct comparison with the in challenge systems impossible, but it still can give an idea of the achievable performances. Again, performances are measured with the BoW and WS representations combined with SBJ and DPLs. As it can be noticed, the use of distributed word representations is beneficial also in this scenario, as demonstrated by the BoW+WS row of Table 8, where a 65.20% in *F1pnn* and 66.35% in *F1pnn* are reported. Again, when using the automatically acquired polarity lexicons, improvements are noticeable, as demonstrated by the 66.11% in the *F1pn* and 67.07% in the *F1pnn* of the BoW+WS+DPLp-EN setting. These are straightforward results if considering that no hand coded resource has been used. Notice that the *Best-System*[11] here reported is measured over the full test set.

5.2. Sentiment Analysis in Twitter in Italian

The impact of the Italian lexicon is measured against the data of the Evalita 2014 Sentipolc (Basile et al., 2014) challenge. Here, Twitter messages are annotated with respect to `subjectivity`, `polarity` and `irony`. We selected those messages annotated with polarity and that were not expressing any irony in order not to have been biased by polarity inversion phenomena typical of ironic texts. Our evaluations are carried out on $2,566$ and $1,175$ messages, used respectively for training and testing.

In Table 9, performances for this setting are reported. Again, the F1 mean between the positive and negative classes (*F1pn*), as well as the mean between all the involved classes (*F1pnn*) are reported.

Kernel	F1pn	F1pnn
BoW	62.49	58.58
BoW+STX	63.50	59.20
BoW+DPL-IT	65.02	59.77
BoW+DPLp-IT	65.38	60.75
BoW+WS	68.26	63.13
BoW+WS+STX	68.46	63.33
BoW+WS+DPL-IT	68.31	63.30
BoW+WS+DPLp-IT	68.28	63.35

Table 9: Twitter Polarity Classification in Italian.

We compare the DPLs with another Italian polarity lexicon, called SENTIX in (Basile and Nissim, 2013). It consists of words automatically annotated with 4 sentiment scores, i.e. *positive*, *negative*, *polarity* and *intensity*. In our evaluation, tweets are described by 4-dimensional vectors, whose scores correspond to sums across all the tweet words (STX representation). In the results, the benefits of using a polarity lexicon for augmenting the BoW representation is more evident, and the improvement in using the two resources is similar. In fact, the BoW kernel alone has a performance of 58.58% in *F1pnn*; when augmented with the STX and the DPLp-IT, the performance increases respectively to 59.20% and 60.75%. Our lexicon is able to provide more information to the learning algorithm, as demonstrated by the higher performance that is measured. When adopting the WS representation, performances increase up to 63.13% in *F1pnn*. When using also the DPLs it seems that the interaction with the WS features is beneficial, as demonstrated by the further improvement up to 63.35% in *F1pnn* with the `DPLp-IT`.

6. Conclusion

In this paper, we acquired large-scale polarity lexicons in English and Italian with a methodology we recently defined in (Castellucci et al., 2015). Emotion related aspects are observed over annotated sentences and are transferred to lexical items. The transfer is made possible as both sentences and words are represented in a common feature space, characterized by a Distributional Model. The method proved to be quite general: it does not rely on any hand-coded resource, but it mainly uses simple cues, e.g. emoticons, for generating a corpus of labeled sentences. Moreover, it is largely applicable to resource-poor languages, e.g. Italian. The generated resources are available for the download to the research community and have been evaluated in different Sentiment Analysis in Twitter tasks.

Future works will investigate how to consider more complex grammatical features in the lexicon generation. The experimented classification algorithms were not sensitive to negation or other grammatical markers or irony phenomena. Moreover, we need to better deal with the neutral class. As experimental findings suggest, the *neutral* items of the lexicon are prevalent, making the acquired lexicon biased towards neutrality. We need a better strategy to select neutral messages to train the corresponding classifier.

[10] The best system measured during the official competition adopted many polarity lexicons and ad-hoc features.

[11] The best system measured during the official competition adopted many polarity lexicons as well as different syntactic (char-ngrams and word-ngrams) and semantic features (word senses and word clusters).

7. References

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proc. of ACL*, pages 238–247. Association for Computational Linguistics.

Basile, V. and Nissim, M. (2013). Sentiment analysis on Italian tweets. In *Proc. of the 4th WS: Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*.

Basile, V., Bolioli, A., Nissim, M., Patti, V., and Rosso, P. (2014). Overview of the evalita 2014 sentiment polarity classification task. In *Proc. of the 4th EVALITA*.

Basili, R. and Zanzotto, F. M. (2002). Parsing engineering and empirical robustness. *Nat. Lang. Eng.*, 8(3):97–120, June.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, March.

Castellucci, G., Croce, D., and Basili, R. (2015). Acquiring a large scale polarity lexicon through unsupervised distributional methods. In Chris Biemann, et al., editors, *Natural Language Processing and Information Systems*, volume 9103 of *LNCS*, pages 73–86. Springer International.

Esuli, A. and Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proc. of 5th LREC*, pages 417–422.

Filice, S., Castellucci, G., Croce, D., and Basili, R. (2015). Kelp: a kernel-based learning platform for natural language processing. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 19–24, Beijing, China, July. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *Processing*, pages 1–6.

Golub, G. and Kahan, W. (1965). Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis*, 2(2):pp. 205–224.

Harris, Z. (1964). Distributional structure. In Jerrold J. Katz et al., editors, *The Philosophy of Linguistics*. Oxford University Press.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proc. of 10th Int. Conf. on Knowledge Discovery and Data Mining*, pages 168–177. ACM.

Kiritchenko, S., Zhu, X., and Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *JAIR*, 50:723–762, Aug.

Landauer, T. and Dumais, S. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. In *Proc. of NAACL*, pages 746–751. Association for Computational Linguistics.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Mohammad, S. M. and Turney, P. D. (2010). Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proc. of CAAGET Workshop*.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *In Proc. of SemEval*, USA, June. ACL.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January.

Rao, D. and Ravichandran, D. (2009). Semi-supervised polarity lexicon induction. In *Proc. of the EACL*, pages 675–682. ACL.

Rosenthal, S., Ritter, A., Nakov, P., and Stoyanov, V. (2014). Semeval-2014 task 9: Sentiment analysis in twitter. In *Proc. SemEval*. ACL and Dublin City University.

Sahlgren, M. (2006). *The Word-Space Model*. Ph.D. thesis, Stockholm University.

Saif, H., Fernandez, M., He, Y., and Alani, H. (2014). Senticircles for contextual and conceptual semantic sentiment analysis of twitter. In Valentina Presutti, et al., editors, *The Semantic Web: Trends and Challenges*, volume 8465 of *LNCS*, pages 83–98. Springer International.

Shawe-Taylor, J. and Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA.

Stone, P. J., Dunphy, D. C., Smith, M. S., and Ogilvie, D. M. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press.

Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proc. of EMNLP*. ACL.

Zhang, Z. and Singh, M. P. (2014). Renew: A semi-supervised framework for generating domain-specific lexicons and sentiment analysis. In *Proc. of 52nd Annual Meeting of the ACL (Volume 1: Long Papers)*, pages 542–551. ACL, June.