Cristina Bosco, Sara Tonelli and Fabio Massimo Zanzotto (dir.)

Proceedings of the Second Italian Conference on Computational Linguistics CLiC-it 2015
3-4 December 2015, Trento

Accademia University Press

A Graph-based Model of Contextual Information in Sentiment Analysis over Twitter

Giuseppe Castellucci, Danilo Croce and Roberto Basili

DOI: 10.4000/books.aaccademia.1332
Publisher: Accademia University Press
Place of publication: Torino
Year of publication: 2015
Published on OpenEdition Books: 11 November 2016
Serie: Collana dell'Associazione Italiana di Linguistica Computazionale
Electronic ISBN: 9788899200008

A Graph-based Model of Contextual Information in Sentiment Analysis over Twitter

Giuseppe Castellucci[(†)], **Danilo Croce**[(‡)], **Roberto Basili**[(‡)]
[(†)]Department of Electronic Engineering
[(‡)]Department of Enterprise Engineering
University of Roma, Tor Vergata
castellucci@ing.uniroma2.it, {croce,basili}@info.uniroma2.it

Abstract

English. Analyzing the sentiment expressed by short messages available in Social Media is challenging as the information when considering an instance is scarce. A fundamental role is played by *Contextual* information that is available when interpreting a message. In this paper, a Graph-based method is applied: a graph is built containing the contextual information needed to model complex interactions between messages. A Label Propagation algorithm is adopted to spread polarity information from known polarized nodes to the others.

Italiano. *Uno dei principali problemi nella analisi delle opinioni nei Social Media riguarda la quantitá di informazione utile che un singolo messaggio puó fornire. Il contesto di un messaggio costituisce un insieme di informazioni utile ad ovviare questo problema per la classificazione della polaritá. In questo articolo proponiamo di rappresentare le interazioni tra i messaggi in grafi che sono poi utilizzati in algoritmi di Label Propagation per diffondere la polaritá tra i nodi.*

1 Introduction

Sentiment Analysis (SA) (Pang and Lee, 2008) faces the problem of deciding whether a text expresses a sentiment, e.g. positivity or negativity. Social Media are observed to measure the sentiment expressed in the Web about products, companies or politicians. The interest in the analysis of tweets led to the definition of highly participated challenges, e.g. (Rosenthal et al., 2014) or (Basile et al., 2014). Machine Learning (ML) approaches are often adopted to classify the sentiment (Pang et al., 2002; Castellucci et al., 2014; Kiritchenko et al., 2014), where specific representations and hand-coded resources (Stone et al., 1966; Wilson et al., 2005; Esuli and Sebastiani, 2006) are used to train some classifier. As tweets are very short, the amount of available information for ML approaches is in general not sufficient for a robust decision. A valid strategy (Vanzo et al., 2014b; Vanzo et al., 2014a) exploits *Contextual* information, e.g. the *reply-to* chain, to support a robust sentiment recognition in online discussions.

In this paper, we foster the idea that Twitter messages belong to a network where complex interactions between messages are available. As suggested in (Speriosu et al., 2011), tweets can be represented in graph structures, along with words, hashtags or users. A Label Propagation algorithm (Zhu and Ghahramani, 2002; Talukdar and Crammer, 2009) can be adopted to propagate (possibly noisy) sentiment labels within the graph. In (Speriosu et al., 2011), it has been shown that such approach can support SA by determining how messages, words, hashtags and users influence each other. The definition of the graph is fundamental for the resulting inference, e.g. when mixing messages about different topics, sentiment detection can be difficult. We take inspiration from the contexts defined in (Vanzo et al., 2014b). In (Speriosu et al., 2011) no explicit relation between messages is considered. We, instead, build a graph where messages in the same context are linked each other and to the words appearing in them. Moreover, we inject prior polarity of words as available in a polarity lexicon (Castellucci et al., 2015). Experiments are carried out over a subset of the Evalita 2014 Sentipolc (Basile et al., 2014) dataset, showing improvements in the polarity classification with respect to not using networked information.

In the remaining, Section 2 presents our graph-based approach. In Section 3 we evaluate the proposed method with respect to a dataset in Italian and we derive the conclusions in Section 4.

2 Sentiment Analysis through Label Propagation over Contextual Graphs

Twitter messages are not created in isolation, but they live in conversations (Vanzo et al., 2014b; Vanzo et al., 2014a). Graph based methods (Zhu and Ghahramani, 2002; Talukdar and Crammer, 2009) provide a natural way to represent tweets in a graph structure in order to exploit relationships between messages to support the SA task.

2.1 Label Propagation Algorithms

In a classification task, given a graph representing a set of objects whose classes are known (labeled seeds) and a set of unlabeled objects, Label Propagation (LP) algorithms spread the label distribution by exploiting the underlying graph. Labels are spread over a graph $\mathbb{G} = \langle V, E, W \rangle$, where V is a set of n nodes, E is a set of edges and W is an $n \times n$ matrix of weights, i.e. w_{ij} is the weight of the edge between nodes i and j.

The Modified Adsorption (MAD) algorithm (Talukdar and Crammer, 2009) is a particular LP algorithm where the spreading of label distributions provides the labeling of all the nodes in the graph, possibly re-labeling also the seeded ones in order to improve robustness against outliers. MAD is defined starting from the Adsorption algorithm (Baluja et al., 2008), where the labeling of all the nodes in a graph is modeled as a controlled random walk. Three actions drive this random walk: `inject` a seeded node with its seed label; `continue` the walk from the current node to a neighbor; `abandon` the walk. These actions are modeled in the MAD algorithm through a minimization problem whose objective function is:

$$\sum_{l \in V} [\, \mu_1 (\vec{Y}_l - \hat{\vec{Y}}_l)^{\mathrm{T}} S (\vec{Y}_l - \hat{\vec{Y}}_l) + \mu_2 \, \hat{\vec{Y}}_l^{\mathrm{T}} L \hat{\vec{Y}}_l \; + \; \mu_3 |\hat{\vec{Y}}_l - R_l| \,] \tag{1}$$

where S, L and R are matrices whose role is to model respectively the relationships between a node and its prior labels, the relationships between two similar nodes and the regularization imposed to the labeling of nodes[1]. The objective function aims at imposing the following constraints to the labeling process with these three terms: the algorithm should assign to a labeled vertex l a distribution $\hat{\vec{Y}}_l$ w.r.t. the target classes that is close to the a-priori distribution ($\vec{Y}_l$); moreover, if two nodes are close according to the graph, then their labeling should be similar. Finally, the third term is a regularization factor. More details about MAD are reported in (Talukdar and Crammer, 2009).

In our approach, the MAD algorithm is applied to a graph where each node is labeled with a distribution over some polarity classes[2]. We assume that a subset of the messages have been annotated, and they are used to train a classifier f that ignore the graph structure. The function f is then used to label the remaining messages so that the MAD algorithm is used to determine the final polarities based on the graph structure.

2.2 Contextual Graph: a definition

In order to generalize the contextual models proposed in (Vanzo et al., 2014b), we build a *Contextual Graph* $\mathbb{G}$ of messages as following. Given a message t_j we consider its context $C(t_j)$ as the list of l preceding messages $t_{j-1}, t_{j-2}, \ldots, t_{j-l}$. The context can be defined as the *reply-to* chain of messages (*conversation* context) or the temporally preceding messages sharing at least one hashtag (*hashtag* context). The contextual graph $\mathbb{G}$ is then built by considering pairs of messages (t_o, t_n) in a context, i.e. $t_o, t_n \in C(t_j)$. These are linked with an edge whose weight w_{t_o,t_n} is computed through a function that depends from the distance between t_o and t_n. In particular, we choose $w_{t_o,t_n} = e^{-\lambda|o-n|}$, where λ controls the influence of messages at different distances. These weighted edges are meant to capture the interaction between close messages in the context. We augment the set of vertices V with nodes representing the words appearing in messages. In particular, given $r_1, r_2, \ldots, r_k$ as the words composing t_o, we add k nodes to V, each representing a word r_i. Each word node is connected to its message and the weight w_{t_o,r_i} is computed through the $\sigma(t_o, r_i)$ function[3]. Word nodes are intended to make the graph connected: without them the graph would be composed by many disconnected sub-graphs, i.e. one per context. Moreover, the

[1] The three hyper-parameters μ_1, μ_2 and μ_3 are used to control the importance of each of these terms.

[2] If a node cannot be initialized with any method, the distribution is initialized with a value of $1/c$, where c is the number of classes.

[3] In the experiments reported below, a boolean function is adopted, i.e. $\sigma(t_o, r_i) = 1$ if r_i belongs to t_o.

more words two messages share, the more they are conveying a similar message. Finally, we define the set of seed nodes as a subset of V that are associated to prior labels. As discussed in the next section, these can be either messages or words: the former are seeded through noisy labels computed from a classification function f; the latter are seeded through label distributions derived from a polarity lexicon.

3 Experimental Evaluation

In order to evaluate the *Contextual Graph* and the MAD algorithm, we adopted a subset of the Evalita 2014 Sentipolc dataset (Basile et al., 2014). It consists of short messages annotated with the `subjectivity`, `polarity` and `irony` classes. We selected those messages annotated with polarity and that were not expressing any ironic content to focus our investigations on less ambiguous messages. Thus, the datasets used for our evaluations consist of a training set Tr of 2,449 messages and a testing set Ts of 1,129 messages.

Dataset	**w/ conv**	**w/ hashtag**	**w/ both**
Tr	349(14,27%)	987(40.36%)	80(3.27%)
Ts	169(14.98%)	468(41.48%)	47(4.16%)

Table 1: Dataset statistics w.r.t. contexts.

As in (Vanzo et al., 2014b), we downloaded the conversation and hashtag contexts that were available at the time of downloading[4]. In Table 1 the number of messages involved in the different contexts are shown. In the experiments, messages are classified with respect to the *positive*, *negative* and *neutral* polarity classes. The message distribution with respect to these classes is shown in Table 2.

Dataset	**positive**	**negative**	**neutral**
Tr	761	973	715
Ts	365	464	300

Table 2: Dataset statistics w.r.t. polarities.

3.1 Graph Construction

In the *Contextual Graph*, vertices represent messages and auxiliary information, such as words. In the LP algorithm each vertex can become a seed, i.e. a distribution w.r.t. the polarity classes can be assigned to it. We first investigate a configuration in which only messages are seeded. Experiments are carried out on three types of *Contextual Graph*s. In the first experiment a graph is built by considering contexts where messages are in a *reply-to* relationship, namely *conversation graph*. A second experiment considers instead the *hashtag* contexts, where messages share at least one hashtag. A third experiment considers both *conversation and hashtag* contexts in the same graph representation. In these configurations, vertices representing words are added to the graph but they are not "seeded" (i.e. they are considered as unlabeled nodes). In the fourth experiment, the last graph is enriched by electing as seeds also words whose sentiment polarity is known a-priori, e.g. derived by a polarity lexicon. In the following, we describe how to associate polarity distributions both to messages and words.

Message seeding. A classification function f that feeds the label distributions for messages is derived by a supervised learning process. In particular, we consider the training set Tr described above, and we acquire a Support Vector Machine multi-classifier in a One-Vs-All schema for the *positive*, *negative* and *neutral* polarity classes as in (Castellucci et al., 2014). Two types of features are adopted: the first is a boolean Bag-of-Words (BOW) feature set. The second is a Wordspace (WS) feature set derived from vector representations of the words in a message, obtained through a neural word embedding (Mikolov et al., 2013). We acquired the embedding from a corpus of 10 million tweets downloaded during the first months of 2015. A *skip-gram* model is acquired through the `word2vec`[5] tool and deriving[6] 250-dimensional vectors for about $99,410$ words. The WS feature set for a message t_j is obtained by considering the linear combination of word vectors that appear in t_j. The SVM classifier realizes the function f that assign the initial label distribution, reflecting the classifier confidence in assigning a polarity to each message, i.e. belonging to both train and test datasets, as well as belonging to contexts.

Words seeding. Seeds words are also considered when building the *Contextual Graph*. In particular, we adopt the Distributional Polarity Lexicon (DPL) (Castellucci et al., 2015) that associates each word to the prior information about the positivity, negativity and neutrality. The lexicon is built as follows: a classifier d is acquired from

[4]Results are not directly comparable to other systems participating to the Evalita 2014 challenge as some message was not more available in Twitter.

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

[6]`word2vec` settings are: min-count=50, window=5, iter=10 and negative=10.

a dataset of generic messages gathered by Twitter considering the occurrence of noisy labels, i.e. emoticons expressing positivity, negativity or neutrality. In a nutshell, given the properties of the vector space WS, we project sentences and words in the same space, in order to transfer the polarity from sentences to words via the classifier d and obtaining the polarity scores of the DPL. The *positivity* and *negativity* scores of a word in DPL are used as seed distribution in the MAD algorithm.

3.2 Experimental Results

A first measure is given by the SVM classifier that is used to assign a polarity distribution to seeds belonging to the test dataset. We measure the mean between the F1 scores of the *positive* and *negative* classes (F1-Pn), and the mean between the F1 scores of all the three classes (F1-Pnn). Different feature representations are exploited in the SVM evaluation, as pointed out in Table 3.

Features	F1-Pn	F1-Pnn
BOW	0.630	0.583
BOW+WS	0.688	0.636

Table 3: SVM results (w/o contexts).

When adopting also the WS features, the performance increases in both the performance measures, with respect to the setting where only BOW features are considered.

Ctx size	F1-Pn	F1-Pnn
3	0.693	0.633
6	0.695	0.634
ALL	0.695	0.637

Table 4: MAD on conversation context.

Ctx size	F1-Pn	F1-Pnn
3	0.696	0.635
6	0.697	0.635
16	0.698	0.634
31	0.701	0.634

Table 5: MAD on hashtag context.

In Tables 4 and 5 the MAD algorithm results[7] are reported w.r.t. the *Conversation* and *Hashtag* contexts, as well to different context sizes, e.g. at size 3 we consider a maximum of 2 messages preceding a target one. The MAD algorithm is able to consistently increase the F1-Pn performance measure, while it is equally performing in the F1-Pnn. When adopting the *Hashtag* context, performances are higher w.r.t. the *Conversation* context setting. This is probably due to the fact the only 15% of the messages belong to a *reply-to* chain, while about 40% of the message belong to a Hashtag context. Moreover, *Hashtag* contexts refer to more coherent sets of messages. It makes the LP algorithm better exploit the graph by assigning similar labeling to nodes in the *Hashtag* context.

In Table 6 we applied the MAD algorithm over a graph built considering both contexts: in this scenario, we tuned and adopted two distinct λ values, i.e. λ_c and λ_h, respectively when weighting messages in conversation and hashtag contexts. Again, the contribution of the contextual information is measured through an increment of both the performance measures. Moreover, the contribution of the two contexts allows to further push the performances up, confirming the need of additional information when dealing with such short messages.

	Message seeding		+DPL	
Ctx Size	F1-Pn	F1-Pnn	F1-Pn	F1-Pnn
3	0.697	0.635	0.703	0.636
6	0.700	0.637	0.705	0.638
16	0.702	0.638	**0.719**	**0.648**
31	0.708	0.640	0.708	0.638

Table 6: MAD on both contexts.

Finally, we injected seed distributions over words through the Distributional Polarity Lexicon (DPL). The lexicon allows injecting a-priori seed on the words in the *Contextual Graph*, resulting in higher performances w.r.t. the case without DPL.

4 Conclusion

In this paper, the *Contextual Graph* is defined as a structure where messages can influence each other by considering both *intra-context* and *extra-context* links: the former are links between messages, while the latter serves to link messages in different contexts through shared words. The application of a Label Propagation algorithm confirms the positive impact of contextual information in the Sentiment Analysis task over Social Media. We successfully injected prior polarity information of words in the graph, obtaining further improvements. This is our first investigation in graph approaches for SA: we only adopted the MAD algorithm, while other algorithms have been defined, since (Zhu and Ghahramani, 2002) and they will be investigated in future works. Moreover, other contextual information could be adopted. Finally, other datasets should be considered, proving the effectiveness of the proposed method that does not strictly depend on the language of messages.

[7] the λ value and the MAD hyper-parameters μ_1, μ_2, μ_3 have been tuned on a validation set in each experiment.

References

Shumeet Baluja, Rohan Seth, D. Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. 2008. Video suggestion and discovery for youtube: Taking random walks through the view graph. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 895–904, New York, NY, USA. ACM.

Valerio Basile, Andrea Bolioli, Malvina Nissim, Viviana Patti, and Paolo Rosso. 2014. Overview of the evalita 2014 sentiment polarity classification task. In *Proc. of the 4th EVALITA*.

Giuseppe Castellucci, Danilo Croce, Diego De Cao, and Roberto Basili. 2014. A multiple kernel approach for twitter sentiment analysis in italian. In *Fourth International Workshop EVALITA 2014*.

Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2015. Acquiring a large scale polarity lexicon through unsupervised distributional methods. In Chris Biemann, Siegfried Handschuh, Andr Freitas, Farid Meziane, and Elisabeth Mtais, editors, *Natural Language Processing and Information Systems*, volume 9103 of *Lecture Notes in Computer Science*, pages 73–86. Springer International Publishing.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of 5th LREC*, pages 417–422.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M. Mohammad. 2014. Sentiment analysis of short informal texts. *JAIR*, 50:723–762, Aug.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

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

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *EMNLP*, volume 10, pages 79–86, Stroudsburg, PA, USA. ACL.

Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. 2014. Semeval-2014 task 9: Sentiment analysis in twitter. In *Proc. SemEval*. ACL and Dublin City University.

Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. 2011. Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the First Workshop on Unsupervised Learning in NLP*, EMNLP '11, pages 53–63, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, and Daniel M. Ogilvie. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press.

Partha Pratim Talukdar and Koby Crammer. 2009. New regularized algorithms for transductive learning. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part II*, ECML PKDD '09, pages 442–457, Berlin, Heidelberg. Springer-Verlag.

Andrea Vanzo, Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2014a. A context based model for sentiment analysis in twitter for the italian language. In Roberto Basili, Alessandro Lenci, and Bernardo Magnini, editors, *First Italian Conference on Computational Linguistics CLiC-it 2014*, Pisa, Italy.

Andrea Vanzo, Danilo Croce, and Roberto Basili. 2014b. A context-based model for sentiment analysis in twitter. In *Proceedings of 25th COLING*, pages 2345–2354. Dublin City University and Association for Computational Linguistics.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of EMNLP*. ACL.

Xiaojin Zhu and Zoubin Ghahramani. 2002. Learning from labeled and unlabeled data with label propagation. Technical report, CMU CALD.