Multimedia Tools and Applications manuscript No.
(will be inserted by the editor)

A Multimodal Feature Learning Approach for Sentiment Analysis of Social Network Multimedia

Claudio Baecchi · Tiberio Uricchio · Marco Bertini · Alberto Del Bimbo

Received: date / Accepted: date

Abstract In this paper we investigate the use of a multimodal feature learning approach, using neural network based models such as Skip-gram and Denoising Autoencoders, to address sentiment analysis of micro-blogging content, such as Twitter short messages, that are composed by a short text and, possibly, an image.

The approach used in this work is motivated by the recent advances in: *i)* training language models based on neural networks that have proved to be extremely efficient when dealing with web-scale text corpora, and have shown very good performances when dealing with syntactic and semantic word similarities; *ii)* unsupervised learning, with neural networks, of robust visual features, that are recoverable from partial observations that may be due to occlusions or noisy and heavily modified images.

We propose a novel architecture that incorporates these neural networks, testing it on several standard Twitter datasets, and showing that the approach is efficient and obtains good classification results.

Keywords Sentiment analysis, feature learning, micro-blogging, Twitter

1 Introduction

In the last few years micro-blogging services, in which users describe their current status by means of short messages, obtained a large success among users. Unarguably, one of the most successful services is Twitter[1], that is used worldwide to discuss about daily activities, to report or comment news, and to share information using messages (called 'tweets') composed by at most

Media Integration and Communication Center (MICC)
Università degli Studi di Firenze, Italy
E-mail: {claudio.baecchi|tiberio.uricchio|marco.bertini|alberto.delbimbo}@unifi.it

[1] Twitter reports to have 271 million monthly active users that send 500 million status updates per day - `https://about.twitter.com/company`

140 characters. Since 2011 Twitter natively supports adding images to tweets, easing the creation of richer content. A study performed by Twitter[2] has shown that adding images to tweets increases user engagement more than adding videos or hashtags.

Despite their brevity these messages often convey also the feeling and the point of view of the people writing them. The addition of images reinforces and clarifies these feelings (see Fig.1). Automatic analysis of the sentiment of these tweets, i.e. retrieving the opinion they express, has received a large attention from the scientific community. This is due to its usefulness in analyzing a large range of domains such as politics [1] and business [2]. Sentiment analysis may encompass different scopes [3]: *i)* polarity, i.e. categorize a sentiment as positive, negative or neutral; *ii)* emotion, i.e. assign a sentiment to an emotional category such as joy or sadness; *iii)* strength, i.e. determine the intensity of the sentiment.

So far, the vast majority of works have addressed only the textual data. In this work we address the classification of tweets, according to their polarity, considering both textual and visual information. We propose a novel schema that, by incorporating a language model based on neural networks, can efficiently exploit web-scale sources corpus and robust visual features obtained from unsupervised learning. The proposed method has been tested on several standard datasets, showing promising results.

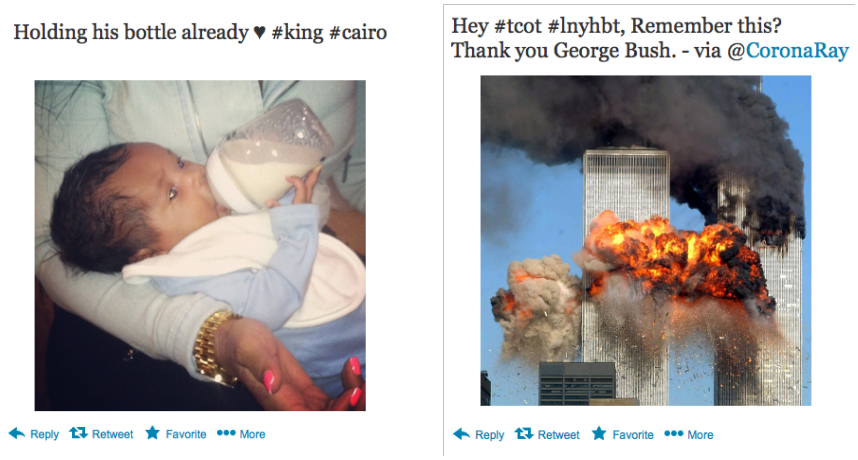

Fig. 1 Examples of tweets with images from the SentiBank Twitter dataset [4]. *left)* positive sentiment tweet; *right)* negative sentiment tweet.

The paper is organized as follows: Sect. 2 provides an overview of previous works; the proposed method is presented in Sect. 3, while experiments on

[2] https://blog.twitter.com/2014/what-fuels-a-tweets-engagement

four standard datasets and comparison with state-of-the-art approaches and baselines are reported in Sect. 4. Conclusions are drawn in Sect. 5.

2 Previous Work

Sentiment analysis in texts. Brevity, sentence composition and variety of topics are among the main challenges in sentiment analysis of tweets (and microblogs in general). In fact these texts are short, often they are not composed carefully as news or product reviews, and cover almost any conceivable topic. Several specific approaches for Twitter sentiment analysis have been proposed, typically using sentence-level classification with n-gram word models. Liu *et al.* [5] concatenate tweets of the same class (polarity) in large documents, from which a language model is derived and then classify tweets through maximum likelihood estimation, using both supervised and unsupervised data for training; the role of unsupervised data is to deal with words that do not appear in the vocabulary that can be built from a small supervised dataset. In [6] three approaches to sentiment classification are compared: Multinomial Naïve Bayes (MNB), Hinge Loss with Stochastic Gradient Descent and Hoeffding Tree; the authors report that MNB outperforms the other approaches. In [7] unigram and bigram features have been used to train Naïve Bayes classifiers, where bigrams help to account for negation of words. Saif *et al.* [8] have evaluated the use of a Max Entropy classifier on several Twitter sentiment analysis datasets. Since using n-grams on tweet data may reduce classification performance due to the large number of infrequent terms in tweets, some authors have proposed to enrich the representation using micro-blogging features such as hashtags and emoticons as in [9], or using semantic features as in [10].

Neural networks language models. Recently, the scientific community has addressed the problem of learning vector representations of words that can represent information like similarity or other semantic and syntactic relations, obtaining better results than using the best n-gram models. The use of neural networks to perform this task is motivated by recent works addressing the scalability of training. In this formulation every word is represented in a distributional space where operations like concatenation and averaging are used to predict other words in context, trained by the use of stochastic gradient descent and backpropagation. In the work of [11], a model is trained based on the concatenation of several words to predict the next word: every word is mapped into a vector space where similar words have similar vector representations. A successive work uses multitask techniques [12] to jointly train several tasks showing improvements in generalization. A fast hierarchical language model was proposed in [13], attacking the main drawback of needing long training and testing times. The use of unsupervised additional words was proposed by [14] showing further improvements using word features learned in advance to a supervised NLP task. Recently Mikolov *et al.* [15] have proposed several improvements on Hierarchical Softmax [13] and Negative Sampling [16]

and introduced the Skip-gram model [17], reducing further the computational cost, and showing fast training on corpora of billions of words [15]. More recently, researchers also extended these models, trying to achieve paragraph and document level representations [18].

Micro-blog multimedia analysis. Most of the works dealing with analysis of the multimedia content of micro-blogs have dealt with content summarization and mining, image classification and annotation. Geo-tagged tweet photos are used in [19, 20] to visually mine events using both textual and visual information. The system presented in [21] provides tools for content curation, creation of personalized web sites and magazines through topic detection of tweets and selection of representative associated multimedia. A system for exploration of events based on facets related to who, when, what, why and how of an event, has been presented in [22], using a Bilateral Correspondence model (BC-LDA) for image and words. A multi-modal extension of LDA has been proposed in [23] to discover sub-topics in microblogs, in order to create a comprehensive summarization.

An algorithm for photo tag suggestion using Twitter and Wikipedia are used in [24] to annotate social media related to events, exploiting the fact that tweets about an event are typically tweeted during its development. Classification of tweets' images in visually-relevant and visually-irrelevant, i.e. images that are correlated or not to the text of the tweet, has been studied in [25], using a combination of text, context and visual features.

Zhao *et al.* [26] have studied the effects of adding multimedia to tweets within Sina Weibo, a Chinese equivalent of Twitter, finding that adding images boosts the popularity of tweets and authors, and extends the lifespan of tweets.

Sentiment analysis in social images. Sentiment analysis of visual data has received so far less attention than that of text data and, in fact, only a few small datasets exist, such as the International Affective Picture System (IAPS) [27] and the Geneva Affective Picture Database (GAPED) [28]. The former provides ratings of emotion (in terms of pleasure, arousal and dominance) for 369 images, while the latter provides 520 images associated to negative sentiment, 89 neutral and 121 positive images. Another related direction is given by works on aesthetics: surveys are provided in [29, 30]. However, none of these datasets deal with social media.

A few works have addressed the problem of multimedia sentiment analysis of social network data. Borth *et al.* [4] have recently presented a large-scale visual sentiment ontology and associated set of detectors, consisting of 3,244 pairs of nouns and adjectives (ANP), based on Plutchik's Wheel of Emotions [31]. Detectors are trained using Flickr images, represented using a combination of global (e.g. color histogram and GIST) and local (e.g. LBP and BoW) features. The paper provides also two publicly available image datasets obtained from Flickr and from Twitter. The system proposed in [32] for the classification of Sina Weibo statuses exploits the ANP detectors proposed in [4], fusing them with text sentiment analysis based on 3 features: *i)* sentiment

words from Hownet (Chinese equivalent to WordNet), *ii)* semantic tags and *iii)* rules of sentence construction, to cope with rhetorical questions, negations and exclamatory sentences.

Cross-media bag-of-words, combining bag of text words with bag of image words obtained from the SentiBank detectors of [4], has been proposed in [33] for sentiment analysis of microblog messages obtained from Sina Weibo. Yang *et al.* [34] have proposed a hybrid link graph for images of social events, weighting links based on textual emotion information, visual similarity and social similarity. A ranking algorithm to discover emotionally representative images in microblog statuses is then presented. The work of Chen *et al.* [35], distinguishes between the intended publisher effect and the sentiment that is induced in the viewer ('viewer affect concept') and aims at predicting the latter. The goals are to recommend appropriate images and suggest image comments.

3 The Proposed Method

Recent works have shown [36] that neural network based language models significantly outperform N-gram models; similarly, the use of neural networks to learn visual features and classify images has shown that they can achieve state-of-the-art results on several standard datasets and international competitions [37]. The proposed method builds on these advances.

We start by describing the well-known text based approach *Continuous Bag-Of-Words* (CBOW) model [17] that is the base of our scheme, then we present our model for polarity classification problem. Finally, we show a further extension of the model to incorporate visual information, based on a Denoising Autoencoder [38], that allows the same unsupervised capabilities on images as CBOW-based methods on text.

3.1 Textual information

Mikolov *et al.* [17] showed that in the CBOW model, words with similar meaning are mapped to similar positions in a vector space. Thus, distances may carry a meaning, allowing to formulate questions in the vector space using simple algebra (e.g. the result of vector('king') - vector('man') + vector('woman') is near vector('queen')). Another property is the very fast training, that allows to exploit large-scale unsupervised corpora such as web sources (e.g. Wikipedia).

Continuous Bag-Of-Words model. In this framework, each word is mapped to a unique vector represented by a column in a word matrix W of Q length. Every column is indexed by a correspondent index from a dictionary V_T. Given a sequence of words $w_1, w_2, \ldots, w_K$, CBOW model with hierarchical softmax aims at maximizing the average log probability of predicting the central word

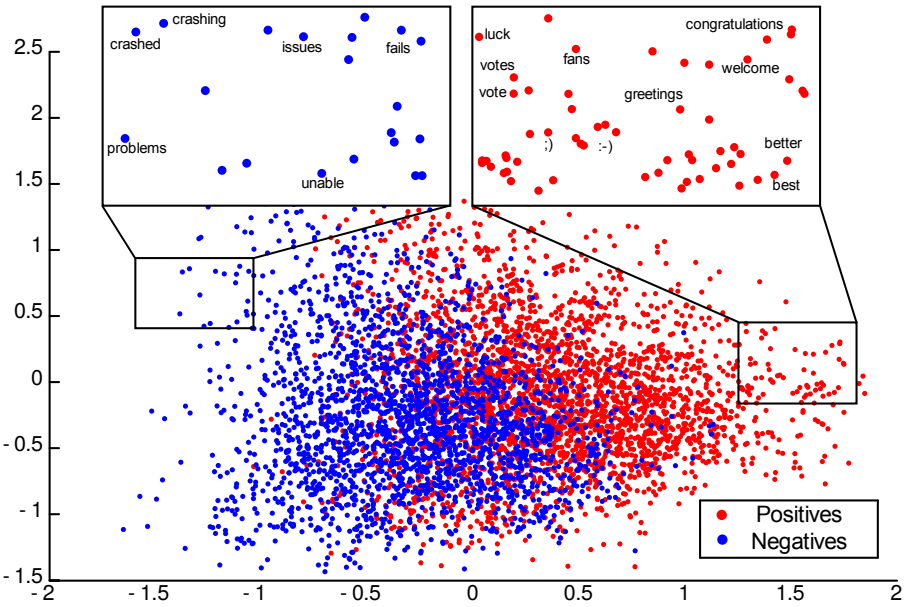

Fig. 2 Visualization of CBOW word vectors trained on tweets of the SemEval-2013 dataset. Blue points are single words classified as negative, while red ones are positive. Semantically similar words are near (e.g. 'crashing' and 'crashed', 'better' and 'best') and share the same polarity.

w_t given the context represented by its M-window of words, i.e. the M words before and after w_t:

$$\frac{1}{K}\sum_{t=M}^{K-M} \log p(w_t|w_{t-M},\ldots,w_{t-1},w_{t+1},\ldots,w_{t+M}) \tag{1}$$

The output $f \in \mathbb{R}^{|V_T|}$ for the model is defined as:

$$f_{w_t} = \left[W_{t-M},\ldots,W_{t-1},W_{t+1},\ldots,W_{t+M}\right]^T G \tag{2}$$

where W_i is the column of W corresponding to the word w_i and $G \in \mathbb{R}^{P\times|V_T|}$. Both W and G are considered as weights and have to be trained, resulting in a dual representation of words. Typically the columns of W are taken as final word features. An output probability is then obtained by using the softmax function on the output of the model:

$$p(w_t|w_{\text{context}}) = \frac{e^{f_{w_t}}}{\sum_i e^{f_{w_i}}} \tag{3}$$

where $w_{\text{context}} = (w_{t-M},\ldots,w_{t-1},w_{t+1},\ldots,w_{t+M})$. When considering a high number of labels, it can be computed more efficiently by employing a hierarchical variation [13], requiring to evaluate $\log_2(|V_T|)$ words instead of $|V_T|$.

In [17], an additional task named *Negative Sampling* is considered, where a word w_l is to be classified as related to the given context or not, i.e. $p(w_l|w_{\text{context}})$:

$$u_{w_l} = \sigma\big(\big[W_{t-M}, \ldots, W_l, \ldots, W_{t+M}\big]^T N_s\big) \tag{4}$$

where $N_s \in \mathbb{R}^Q$ and σ is the logistic function. Depending on w_l as the actual w_t word or a randomly sampled one, u_{w_l} has a target value of respectively 1 or 0.

The CBOW-LR method. Our model, denoted as CBOW-LR, is an extension of CBOW with negative sampling, specialized on the task of sentiment classification. An important difference from approaches that directly use a CBOW representation, or from [14], is that our model learns representation and classification concurrently. Considering that multi-task learning can improve neural networks performance [14], the idea is to use two different contributions accounting for semantic and sentiment polarity, respectively.

Given a corpus of tweets $\mathbf{X}$ where each tweet is a sequence of words $w_1, w_2, \ldots, w_K$, we aim at classifying tweets as positive or negative, and learn word vectors $W \in \mathbb{R}^{Q \times |V_T|}$ with properties related to the sentiment carried by words, while retaining semantic representation. Semantic representation can be well-represented by a CBOW model, while sentiment polarity has limited presence or is lacking. Note that polarity supervision is limited and possibly weak, thus a robust semi-supervised setting is preferred: on the one hand, a model of sentiment polarity can use the limited supervision available, on the other hand the ability to exploit a large corpus of unsupervised text, like CBOW, can help the model to classify previously unseen text. This is explicitly accounted in our model by considering two different components:

i) inspired by [17], we consider a feature learning task on words by classifying sentiment polarity of a tweet. A tweet is represented as a set of M-window of words that we denote as $\mathcal{G}$. Each window $\mathcal{G}$ is represented as a sum of their associated word vectors W_i, and a polarity classifier based on logistic regression is applied accordingly:

$$y(\mathcal{G}) = \sigma(C^T(\sum_{W_i \leftarrow w_i \in \mathcal{G}} W_i) + b_s) \tag{5}$$

Here the notation $W_i \leftarrow w_i \in \mathcal{G}$ refers to selecting the i-th column of W by matching the w_i word from $\mathcal{G}$. The matrix $C \in \mathbb{R}^Q$ and the vector $b_s \in \mathbb{R}$ are parameters of a logistic regression, while a binary cross entropy is applied as loss function for every window $\mathcal{G}$. This is applied for every tweet T labeled with $\overline{y}_T$ in the training set and results in the following cost:

$$C_{\text{sent}} = \sum_{(T,\overline{y}_T)} \sum_{\mathcal{G} \in T} -\overline{y}_T \log(y(\mathcal{G}) - (1 - \overline{y}_T) \log(1 - y(\mathcal{G}))) \tag{6}$$

However, differently from a standard logistic regression, the representation matrix W is also a parameter to be learned. A labeled sentiment dataset is required to learn this task.

ii) we explicitly represent semantics by adding a task similar to negative sampling, without considering the hierarchical variation. The idea is that a CBOW model may also act as a regularizer and provide an additional semantic knowledge of word context. Given a window $\mathcal{G}$, a classifier has to predict if a word w_l fits in it. To this end, an additional cost is added:

$$C_{\text{sem}} = \sum_{T} \sum_{\mathcal{G} \in T} \sum_{(r_l, w_l) \in \mathcal{F}} (r_l - u_{w_l})^2 \tag{7}$$

where $\mathcal{F}$ is a set of words w_l with their associated target r_l, derived from a training text sequence. This is the core of negative sampling: $\mathcal{F}$ always contains the correct word w_t for the considered context $\mathcal{G}$ ($r_l = 1$) and $K - 1$ random sampled words from V_T ($r_l = 0$). It is indeed a sampling as $K < |V_T| - 1$ of the remain wrong words. Note that differently from the previous task, this is unsupervised, not requiring labeled data; moreover tweets can belong to a different corpus than that used in the previous component. This allows to perform learning on additional unlabeled corpora, to enhance word knowledge beyond that of labeled training words.

Finally, concurrent learning is obtained by forging a total cost, defined by the sum of the two parts, opportunely weighted by a $\lambda \in [0, 1]$, and minimized with SGD:

$$C_{\text{CBOW-LR}} = \lambda \cdot C_{\text{sent}} + (1 - \lambda) \cdot C_{\text{sem}} \tag{8}$$

Fig. 2 visualizes the word vectors learned by our model. Note the tendency of separating the opposite polarities and the fact that similar words are close to each other.

At prediction time, for each word in a tweet T we consider its M-window $\mathcal{G}$ and we compute (5) for each window, summing the results:

$$Pred(T) = \sum_{\mathcal{G} \in T} \Big(y(\mathcal{G}) - 0.5 \Big) \tag{9}$$

If $Pred(T) < 0$ the tweet is labeled as negative, otherwise it is considered positive. It is worth noticing that at prediction time the method does not consider a word as positive or negative in its own, but it uses also its context to classify its sentiment and how strong it is. Thus the same word can be classified differently if used in different contexts.

3.2 Textual and Visual Information

The CBOW-LR model presented in Sect. 3.1 can be extended to account for visual information, such as that of images associated to tweets or status messages. Popular image representations are the Visual Bag-Of-Words Model [39, 40, 41], Fisher Vector [42] and its improved version [43, 44]. However, as shown recently in [37, 45], neural network based models have been shown to widely outperform these previous models. So, to fit with the CBOW representation discussed in the previous section, we choose to exploit the images by

using a representation similar to the one used for the textual information, i.e. a representation obtained from the whole training set by means of a neural network. Moreover, likewise for the text, unsupervised learning can be performed. For these reasons, inspired also by works such as [38], we choose to extend our network with a single-layer Denoising Autoencoder, to take its middle level representation as our image descriptor. As for the textual version, the inclusion of this additional task allows our method to concurrently learn a textual representation and a classifier on text polarity and its associated image.

Denoising Autoencoder. In general, an Autoencoder (also called Autoassociator [46]) is a kind of neural network trained to encode the input into some representation (usually of lower dimension) so that the input can be reconstructed from that representation. For this type of network the output is thus the input itself. Specifically, an Autoencoder is a network that takes as input a K-dimensional vector x and maps it to a hidden representation h through the mapping:

$$h = \sigma(P_e \; x + b_e) \tag{10}$$

where σ is the sigmoid function (but any other non-linear activation function can be used), P_e and b_e are respectively a matrix of encoding weights and a vector of encoding biases. At this point, h is the coded representation of the input, and has to be mapped back to x. This second part is called the reconstruction z of x (being z of the same dimension and domain of x). In this step a similar transformation as in Eq. 10 is used:

$$z = \sigma(P_d \; h + b_d) \tag{11}$$

where P_d and b_d are respectively a matrix of decoding weights and a vector of decoding biases. One common choice is to constrain $P_d = P_e^T$; in this configuration the Autoencoder is said to have 'tied weights'. The motivation for this is that tied weights are used as a regularizer, to prevent the Autoencoder to learn the identity matrix when the dimension of the hidden layer is big enough to memorize the whole input; another important advantage is that the network has to learn fewer parameters. With this configuration, Eq. (11) becomes:

$$\hat{z} = \sigma(P_e^T \; h + b_d) \tag{12}$$

Learning is performed by minimizing the cross-entropy between the input x and the reconstructed input z:

$$L(x, z) = -\sum_{k=1}^{K} \Big(x_k \; \log z_k + (1 - x_k) \; \log \; (1 - z_k) \Big) \tag{13}$$

using stochastic gradient descent and backpropagation.

In this scenario h is similar to a lossy compression of x, that should capture the coordinates along the main directions of variation of x. To further improve the network, the input x can be 'perturbed' to another slightly different image, $\tilde{x}$, so that the network will not adapt too much to the given inputs but will be

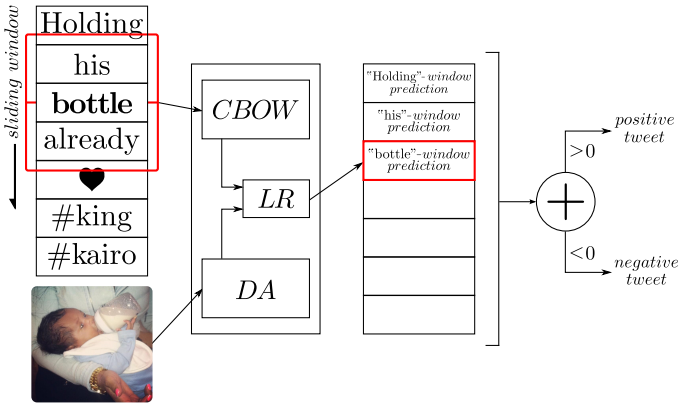

Fig. 3 The process of polarity prediction of a tweet with its associated image performed by our model. On the left, one tweet text window (in red) at a time is fed into the CBOW model to get a textual representation. Likewise, the associated image is fed into the denoising autoencoder (DA). The two representations are concatenated and a polarity score for the window is obtained from the logistic regression (LR). Finally, each window polarity is summed into a final tweet polarity score.

able to better generalize over new samples. This forms the Denoising variant of the Autoencoder. To do this, the input is corrupted by randomly setting some of the values to zero [46]. This way the Denoising Autoencoder will try to reconstruct the image including the missing parts. Another benefit of the stochastic corruption is that, when using a hidden layer bigger than the input layer, the network does not learn the identity function (which is the simplest mapping between the input and the output) but instead it learns a more useful mapping, since it is trying to also reconstruct the missing part of the image.

The CBOW-DA-LR method. The model used to deal with textual and visual information, denoted as CBOW-DA-LR, is an extension of CBOW-LR with the addition of a new task based on a Denoising Autoencoder (DA) applied to images, aiming at obtaining a mid-level representation. In this final form, the descriptor obtained from the DA, together with the continuous word representation, represents the new descriptor for a window of words in a tweet and is concurrently used to learn a logistic regressor. Given a tweet, for each window, we compute the continuous word representation and the image descriptor associated with the tweet. Each window in a tweet will be associated with the same image descriptor as the image for the tweet is always the same.

Fig. 3 shows an exemplification of the prediction process for a tweet with its accompanying image. While the image gets a fixed representation for the entire process, the text is represented one window at a time through a sliding window process. Each window is processed independently to get a local polarity score. To get the overall tweet polarity, each window polarity is summed into a final score and classified according to its sign.

This can be formalized as follows: if we define $h_{\mathcal{G}}$ as the encoding of the image associated to the window $\mathcal{G}$ of the tweet T, then Eq. (5) becomes:

$$y(\mathcal{G}) = \sigma\left(C^T\Big(\big(\sum_{W_i \leftarrow w_i \in \mathcal{G}} W_i\big) \parallel (h_{\mathcal{G}})\Big) + b_s\right) \tag{14}$$

where $\parallel$ is the concatenation operator, i.e. the encoded representation of the image is concatenated to the continuous word representation of the window, forming a new vector whose size is the sum of the size of the continuous word space and the size of the encoding representation of the image.

As stated before, the Autoencoder can be pre-trained in the same fashion as the continuous word representation. Any set of unlabeled images can be used to train the network before the actual training on the tweets.

The DA will be a component of our model and, like the two previous components CBOW and LR, it has its own cost function. Similar to Eq. (13), it is:

$$C_{\text{image}} = -\sum_{k=1}^{K}\Big(\tilde{x}_k \ \log \hat{z}_k + (1-\tilde{x}_k) \ \log\ (1-\hat{z}_k)\Big) \tag{15}$$

Since we are aiming at concurrent learning the textual and image representations, the three components are combined together in a single final cost of CBOW-DA-LR. Starting from the previously defined Eq. (8) for CBOW and Eq. (7) for LR, the cost becomes:

$$C_{\text{CBOW-DA-LR}} = \lambda_1 \cdot C_{\text{sent}} + \lambda_2 \cdot C_{\text{sem}} + \lambda_3 \cdot C_{\text{image}} \tag{16}$$

where $\lambda_1, \lambda_2, \lambda_3$ weight the contribution of each task. The model can be trained by minimizing $C_{\text{CBOW-DA-LR}}$ with stochastic gradient descend. Symbolic derivatives can be easily obtained by using an automatic differentiation algorithm (e.g. Theano [47]). After training, Eq. (9) can be used to predict the label of the tweet in the same manner as it is used when we do not consider the image descriptor.

4 Experiments

The datasets. To evaluate the proposed approach we have used four datasets obtained from Twitter:

i) Sanders Corpus[3], consists of 5,513 manually labelled tweets on 4 topics (Apple, Google, Microsoft and Twitter). Of these, after removing missing tweets, retweets ad duplicates, only 3,625 remain. The dataset does not specify a train and a test subset, so to evaluate the performance the whole set is randomly divided multiple times into subsets each time each one with the same size and the mean performance is reported;

[3] `http://sananalytics.com/lab/twitter-sentiment/`

ii) Sentiment140[4] [48] consists of a 1.6 million tweet training set collected and weakly annotated by querying positive and negative emoticons, considering a tweet positive if it contains a positive emoticon like " :) "and negative if, likewise, it contains a negative emoticon like " :("; the dataset also comprises a manually annotated test set of 498 tweets obtained querying names of products, companies and people;

iii) SemEval-2013[5] provides a training set of 9,684 tweets of which only 8,208 are not missing at the time of writing and a test set of 3,813 tweets, selected querying a mixture of entities, products and events; the dataset is part of the SemEval-2013 challenge for sentiment analysis and also comprises of a development set of 1,654 (of which only 1,413 available at the time of writing) that can be used as an addendum to the training set or as a validation set;

iv) SentiBank Twitter Dataset[6], consists of 470 positive and 133 negative tweets with images, related to 21 topics, annotated using Mechanical Turk; the dataset has been partitioned by the authors into 5 subsets, each of around 120 tweets with the respective images, to be used for a 5-fold cross-validation.

In this work we consider the binary positive/negative classification, thus we have removed neutral/objective tweets from the corpora when necessary. This approach follows that of [48] and [5], and is motivated by the difficulty to obtain training data for this class; it has to be noted that even human annotators tend to disagree whether a tweet has a negative/positive polarity or it is neutral [49]. Performance is reported in terms of Accuracy. The evaluation for SemEval is performed using F_1, since this is the metric originally used in this dataset.

For the Sanders dataset, as described earlier, there is no definition of an actual test set nor of a training set. For these reasons we choose to follow the experimental setup of [5], where experiments on Sanders dataset have been performed varying the number of training tweets between 32 to 768. For each test, first the number of training tweets is selected, then half of them are randomly chosen from all the positive tweets and the other half are chosen from the negative ones. Finally, the remaining tweets are used as test set. Since there could be some variation from a random set to another, for each test 10 different runs are evaluated and the mean is taken as the result of the selected test. Results with this dataset are reported with the notation "Sanders@n", where n is the number of training tweets selected.

The evaluation of the SentiBank dataset has been performed preserving the structure given by the authors so that the results could be comparable. The dataset is divided into 5 subsets for 5-fold cross-validation. Each at a time a subset is considered as test set while the other 4 are considered as training set; 5 runs are performed and in the end the mean of the 5 results is computed and considered the resulting value given by the method for the

[4] http://help.sentiment140.com/for-students

[5] http://www.cs.york.ac.uk/semeval-2013/task2/

[6] http://www.ee.columbia.edu/ln/dvmm/vso/download/sentibank.html

dataset. Considering the high imbalance between positive and negative tweets of this dataset we report also the F_1 score in addition to Accuracy.

We have evaluated the proposed method through a set of 5 experiments: in the first one we evaluate the performance of the proposed CBOW-LR text model comparing it against the standard CBOW model. Then we assess the performance of these models after pre-training them with large scale Twitter corpora. In a third experiment we compare the proposed approach against a baseline and two state-of-the-art methods. In the final experiment we compare the proposed CBOW-DA-LR text+image model against a state-of-the-art method on a publicly available dataset composed by tweets with images. In all these experiments we empirically fixed $K = 5$ and $Q = 100$. In the last experiment we evaluate the effects of K and Q parameters w.r.t. the classification performance an all the datasets. Regarding λ in the first three experiments and $\lambda_1, \lambda_2, \lambda_3$ in the last one, we tested several combinations and found a good setting by fixing $\lambda = 0.5$ and $\lambda_1 = \lambda_2 = \lambda_3 = 0.33$, respectively. Also the image DA was implemented with 'tied weights' to reduce overfitting. Its dimensionality was tested in the range $[200, 1000]$ and found it better performing by fixing it to 500. To perform the optimization using stochastic gradient descent, we employed Theano [47] to automatically compute the derivatives.

Exp. 1: Comparison with baselines. Tab. 1 compares our proposed method (CBOW-LR) with two baselines: RAND-LR and CBOW+SVM. The purpose is twofold: *i)* since we are learning features crafted for the specific task, we compare our method with randomly generated features. RAND-LR learns a logistic regression classifier on random word features (i.e. we set $\lambda = 1$ in eq. 8); *ii)* we verify the superiority of CBOW-LR learned features against a standard unsupervised CBOW representation. The CBOW+SVM baseline employs SVM with standard pre-trained CBOW representation on the specific dataset.

Performance figures show that the proposed method consistently outperforms both baselines, thus our method learns useful representations with some improvement over CBOW.

Exp. 2: Exploiting CBOW training on large scale data. Tab. 2 compares our proposed method with two baselines when exploiting large scale training data for the CBOW representation. We pre-trained a CBOW model using the 1.6 million tweets of Sentiment140 and used the learned features (termed CBOW_S) with two standard learning algorithms. CBOW_S+LR employs the logistic regression while CBOW_S+SVM uses the SVM classifier. In contrast to the baselines, our model CBOW_S-LR employs the pre-trained CBOW_S features as initialization for the W matrix. Comparing Tab. 2 with Tab. 1 shows that CBOW_S+SVM baseline benefit from the use of pre-learned CBOW_S. This is visible especially on the Sanders dataset, as more rich representation is built. Note that when CBOW_S+SVM is applied to Sentiment140 dataset it corresponds to CBOW+SVM, since CBOW_S description is trained on Sentiment140; therefore the result is the same.

Dataset	(proposed) CBOW-LR	RAND-LR	CBOW+SVM
Sentiment140	83.01	61.56	79.39
SemEval-2013 (F_1)	72.57	53.01	71.32
Sanders @ 32	62.55	58.38	59.89
Sanders @ 256	74.91	63.69	67.91
Sanders @ 768	82.69	65.53	73.03

Table 1 Comparison between our method and two baselines. Performance is reported in terms of accuracy except for SemEval-2013, where is used the F_1 measure. Sanders@n indicates the number of training tweets used for the experiments on that dataset.

Dataset	(proposed) CBOW_S-LR	CBOW_S+LR	CBOW_S+SVM
Sentiment140	83.84	76.32	79.39
Semeval-2013 (F_1)	72.23	73.73	71.48
Sanders @ 32	66.28	66.90	66.65
Sanders @ 256	76.33	71.14	73.69
Sanders @ 768	82.98	75.43	76.44

Table 2 Comparison between our method and two baselines, using an initialization based on CBOW pre-trained aside with 1.6 million tweets of Sentiment140. Performance is reported in terms of accuracy except for SemEval-2013, where is used the F_1 measure. Sanders@n indicates the number of training tweets used for the experiments on that dataset.

While both CBOW_S+SVM and CBOW_S+LR are unable to modify the word vector representation, our model CBOW_S-LR is able to retain the full richness of the initial representation and improve it on two datasets.

Exp. 3: Comparison with FSLM and ESLAM. In this experiment we have compared both textual variants of our approach, one with CBOW trained using the dataset on which the method is applied and one using CBOW_S, with two state-of-the-art methods: FSLM and ESLAM, proposed in [5]. FSLM uses a fully supervised probabilistic language model, learned concatenating all the tweets of the same class to form synthetic documents. ESLAM extends FSLM exploiting noisy tweets, based on the presence of 'positive' and 'negative' emoticons, to smooth the language model. Inclusion of manually labelled data with the unsupervised noisy data gives the power to deal with unforeseen text that is not easily handled by fully supervised methods. Fig. 4 shows the Accuracy while varying the number of training tweets of the Sanders dataset. The proposed approach has a much lower performance when using only 32

or 64 tweets for training. However, it can be observed that as the number of training data increases so does the performance of the proposed method, that outperforms that of ESLAM when using 768 tweets for training. In general the proposed method outperforms FSLM. The fact that ESLAM outperforms the proposed method when using smaller training data can be explained by the fact that CBOW models, as Skip-Gram and feature learning methods, require large training datasets.

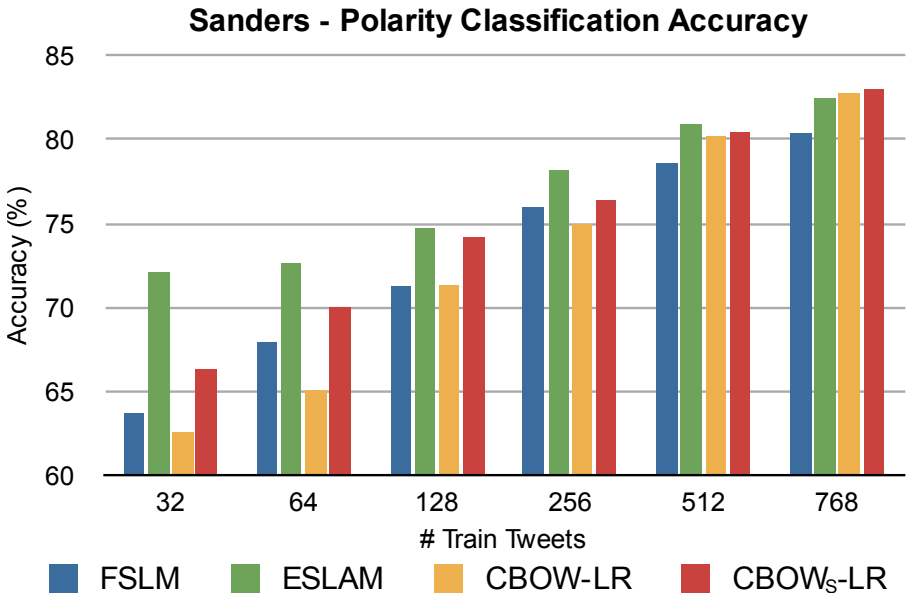

Fig. 4 Comparison between our method with FSLM and ESLAM [5] on Sanders dataset, while varying the number of training tweets.

Exp. 4: Exploiting textual and visual data. In this experiment we have evaluated the performance of three versions of our proposed approach – CBOW-LR for text, DA-LR for visual data, and CBOW-DA-LR for both text and visual information – with different baselines and state-of-the-art approaches.

CBOW-LR has been compared with SentiStrenght [50] and the CBOW+SVM baseline used in Exp. 1 and Exp. 2. DA-LR has been compared with SentiBank [4] classifiers. CBOW-DA-LR has been compared with the approach proposed by the authors of the SentiBank Twitter dataset [4], that uses SentiStrenght [50] API[7] for text classification and SentiBank classifiers as mid-level visual features, with a logistic regression model. As the dataset is imbalanced, we also compare these approaches with an additional baseline based on random classification, i.e. we assign a random polarity to each test tweet. We used the code provided by the authors of the methods, except for the

[7] http://sentistrength.wlv.ac.uk/

Data	Method	SentiBank (AC)	SentiBank (F_1)
	Random	47	42
Text	SentiStrenght [50]	58	51
	CBOW+SVM	72	50
	(proposed) CBOW-LR	75	52
Image	SentiBank [4]	71	51
	(proposed) DA-LR	69	51
Text+Image	SentiStrenght [50] + SentiBank [4]	72	n.a.
	(proposed) CBOW-DA-LR	**79**	**57**

Table 3 Comparison between our method (on single and combined modalities) with baselines and state-of-the-art approaches on SentiBank Twitter Dataset.

SentiStrenght+SentiBank case, for which we report the result published in [4]. Results reported in Tab. 3 show that not only CBOW-LR outperforms both the baseline and SentiStrenght, but also the multimodal SentiStrenght+SentiBank approach. When using only visual information SentiBank obtains a better performance than DA-LR. Considering the text+image case it can be observed that the proposed multimodal CBOW-DA-LR method improves upon single modalities (CBOW-LR and DA-LR) and outperforms SentiStrenght+SentiBank by a larger margin, proving that images hold meaningful informations regarding the polarity of text, and thus can be exploited to improve overall Accuracy and F_1.

Exp. 5: Parameters analysis. Fig. 5 shows accuracy and F_1 of our model when varying K and Q parameters on Sanders, SemEval-2013 and Sentiment140 datasets. The performance on SentiBank is practically not affected by these parameters. The same set of parameters results in the best performance on all the datasets. The values of K and Q are in line with those obtained to train CBOW models on Wikipedia by Mikolov *et al.* .

5 Conclusions

In this paper we have presented a method for sentiment analysis of social network multimedia, presenting an unified model that considers both textual and visual information.

Regarding textual analysis we described a novel semi-supervised model CBOW-LR, extending the CBOW model, that learns concurrently vector representation and a sentiment polarity classifier on short texts such as that of

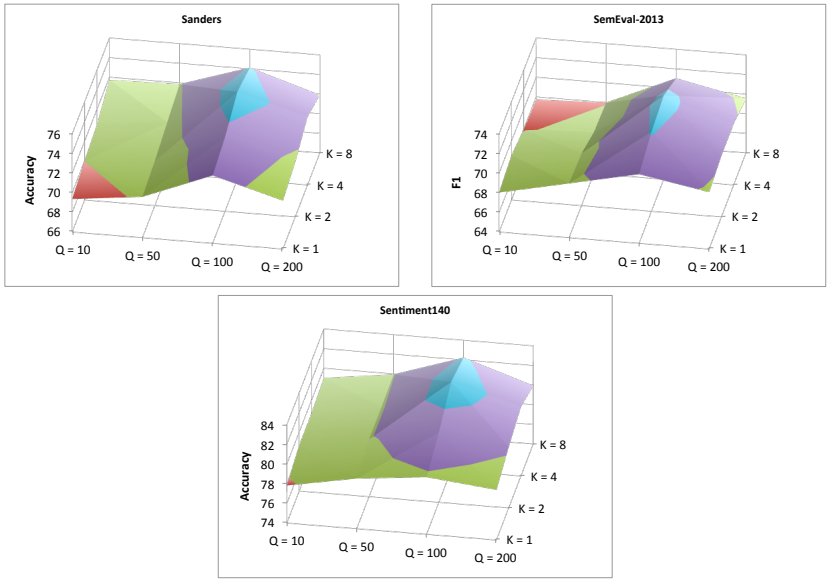

Fig. 5 Performance of the proposed method when varying K and Q parameters on Sanders, SemEval-2013 and Sentiment140 datasets.

tweets. Our experiments show that CBOW-LR can obtain improved accuracy on polarity classification over CBOW representation on the same quantity of text. When considering a large unsupervised corpus of tweets as additional training data for CBOW, a further improvement is shown, with our model being able to improve the overall accuracy. Comparison with the state-of-the-art methods FSLM and ESLAM shows promising results.

The CBOW-LR model has been expanded to account for visual information using a Denoising Autoencoder. The unified model (CBOW-DA-LR) works in an unsupervised and semi-supervised manner, learning text and image representation, as well as the sentiment polarity classifier for tweets containing images. The unified CBOW-DA-LR model has been compared with SentiBank, a state-of-the-art approach on a publicly available Twitter dataset, obtaining a higher classification accuracy.

References

1. Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M.: Predicting elections with Twitter: What 140 characters reveal about political sentiment. In: Proc. of AAAI International Conference on Weblogs and Social Media (ICWSM) (2010)
2. Ghiassi, M., Skinner, J., Zimbra, D.: Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural net-

work. Expert Systems with Applications **40**(16), 6266–6282 (2013). DOI 10.1016/j.eswa.2013.05.057
3. Bravo-Marquez, F., Mendoza, M., Poblete, B.: Combining strengths, emotions and polarities for boosting Twitter sentiment analysis. In: Proc. of ACM International Workshop on Issues of Sentiment Discovery and Opinion Mining (WISDOM) (2013). DOI 10.1145/2502069.2502071
4. Borth, D., Ji, R., Chen, T., Breuel, T., Chang, S.F.: Large-scale visual sentiment ontology and detectors using adjective noun pairs. In: Proc. of ACM International Conference on Multimedia (MM), pp. 223–232 (2013). DOI 10.1145/2502081.2502282
5. Liu, K.L., Li, W.J., Guo, M.: Emoticon smoothed language models for Twitter sentiment analysis. In: Proc. of AAAI Conference on Artificial Intelligence (CAI) (2012)
6. Bifet, A., Frank, E.: Sentiment knowledge discovery in Twitter streaming data. In: Proc. of International Conference on Discovery Science (DS) (2010). DOI 10.1007/978-3-642-16184-1_1
7. Deitrick, W., Hu, W.: Mutually enhancing community detection and sentiment analysis on Twitter networks. Journal of Data Analysis and Information Processing **1**(3), 19.29 (2013)
8. Saif, H., Fernandez, M., He, Y., Alani, H.: Evaluation datasets for Twitter sentiment analysis. In: Proc. of AI*IA Emotion and Sentiment in Social and Expressive Media (ESSEM) (2013)
9. Barbosa, L., Feng, J.: Robust sentiment detection on Twitter from biased and noisy data. In: Proc. of International Conference on Computational Linguistics (COLING) (2010)
10. Saif, H., He, Y., Alani, H.: Semantic sentiment analysis of Twitter. In: Proc. of International Conference on the Semantic Web (ISWC) (2012)
11. Bengio, Y., Schwenk, H., Senécal, J.S., Morin, F., Gauvain, J.L.: Neural probabilistic language models. In: Innovations in Machine Learning, pp. 137–186. Springer (2006)
12. Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proc. of International Conference on Machine Learning (ICML) (2008)
13. Mnih, A., Hinton, G.E.: A scalable hierarchical distributed language model. In: Proc. of Neural Information Processing Systems (NIPS) (2009)
14. Turian, J., Ratinov, L., Bengio, Y.: Word representations: a simple and general method for semi-supervised learning. In: Proc. of ACL Annual Meeting of the Association for Computational Linguistics (2010)
15. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proc. of Neural Information Processing Systems (NIPS) (2013)
16. Gutmann, M.U., Hyvärinen, A.: Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. The Journal of Machine Learning Research (JMLR) **13**(1), 307–361 (2012)
17. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)

18. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: Proc. of International Conference on Machine Learning (ICML) (2014)
19. Yanai, K.: World Seer: A realtime geo-tweet photo mapping system. In: Proc. of ACM International Conference on Multimedia Retrieval (ICMR), pp. 65:1–65:2 (2012). DOI 10.1145/2324796.2324870
20. Kaneko, T., Harada, H., Yanai, K.: Twitter visual event mining system. In: Proc. of IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 1–2 (2013). DOI 10.1109/ICMEW.2013.6618224
21. Serra, G., Alisi, T., Bertini, M., Ballan, L., Del Bimbo, A., Goix, L., Licciardi, C.: STAMAT: A framework for social topics and media analysis. In: Proc. of IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 1–2 (2013). DOI 10.1109/ICMEW.2013.6618227
22. Wang, Z., Cui, P., Xie, L., Chen, H., Zhu, W., Yang, S.: Analyzing social media via event facets. In: Proc. of ACM International Conference on Multimedia (MM), pp. 1359–1360 (2012). DOI 10.1145/2393347.2396484
23. Bian, J., Yang, Y., Chua, T.S.: Multimedia summarization for trending topics in microblogs. In: Proc. of the ACM International Conference on Information & Knowledge Management (CIKM), pp. 1807–1812 (2013). DOI 10.1145/2505515.2505652
24. McParlane, P.J., Jose, J.: Exploiting Twitter and Wikipedia for the annotation of event images. In: Proc. of ACM SIGIR Interantional Conference on Research & Development in Information Retrieval, pp. 1175–1178 (2014). DOI 10.1145/2600428.2609538
25. Chen, T., Lu, D., Kan, M.Y., Cui, P.: Understanding and classifying image tweets. In: Proc. of ACM International Conference on Multimedia (MM), pp. 781–784 (2013). DOI 10.1145/2502081.2502203
26. Zhao, X., Zhu, F., Qian, W., Zhou, A.: Impact of multimedia in Sina Weibo: Popularity and life span. In: Proc. of Chinese Semantic Web Symposium and the First Chinese Web Science Conference (CSWS & CWSC) (2012)
27. Lang, P.J., Bradley, M.M., Cuthbert, B.N.: International affective picture system (iaps): Technical manual and affective ratings (1999)
28. Dan-Glauser, E., Scherer, K.: The geneva affective picture database (gaped): a new 730-picture database focusing on valence and normative significance. Behavior Research Methods **43**(2), 468–477 (2011). DOI 10.3758/s13428-011-0064-1
29. Wang, W., He, Q.: A survey on emotional semantic image retrieval. In: Proc. of IEEE International Conference on Image Processing (ICIP), pp. 117–120 (2008). DOI 10.1109/ICIP.2008.4711705
30. Joshi, D., Datta, R., Fedorovskaya, E., Luong, Q.T., Wang, J., Li, J., Luo, J.: Aesthetics and emotions in images. IEEE Signal Processing Magazine (MSP) **28**(5), 94–115 (2011). DOI 10.1109/MSP.2011.941851
31. Plutchik, R.: The nature of emotions. American Scientist **89**(4), 344–350 (2001)

32. Cao, D., Ji, R., Lin, D., Li, S.: A cross-media public sentiment analysis system for microblog. Multimedia Systems (MS) pp. 1–8 (2014). DOI 10.1007/s00530-014-0407-8
33. Wang, M., Cao, D., Li, L., Li, S., Ji, R.: Microblog sentiment analysis based on cross-media bag-of-words model. In: Proc. of International Conference on Internet Multimedia Computing and Service (ICIMCS), pp. 76:76–76:80 (2014). DOI 10.1145/2632856.2632912
34. Yang, Y., Cui, P., Zhu, W., Zhao, H.V., Shi, Y., Yang, S.: Emotionally representative image discovery for social events. In: Proc. of ACM International Conference on Multimedia Retrieval (ICMR), pp. 177:177–177:184 (2014). DOI 10.1145/2578726.2578749
35. Chen, Y.Y., Chen, T., Hsu, W.H., Liao, H.Y.M., Chang, S.F.: Predicting viewer affective comments based on image content in social media. In: Proc. of ACM International Conference on Multimedia Retrieval (ICMR), pp. 233:233–233:240 (2014). DOI 10.1145/2578726.2578756
36. Mikolov, T., Deoras, A., Kombrink, S., Burget, L., Cernocky, J.H.: Empirical evaluation and combination of advanced language modeling techniques. In: Proc. of Interspeech (2011)
37. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proc. of Neural Information Processing Systems (NIPS), pp. 1097–1105 (2012)
38. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proc. of International Conference on Machine Learning (ICML), pp. 1096–1103 (2008). DOI 10.1145/1390156.1390294
39. Grauman, K., Darrell, T.: The pyramid match kernel: Discriminative classification with sets of image features. In: Proc. of International Conference on Computer Vision (ICCV) (2005)
40. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proc. of Conference on Computer Vision and Pattern Recognition (CVPR) (2006)
41. Li, T., Mei, T., Kweon, I.S., Hua, X.S.: Contextual bag-of-words for visual categorization. IEEE Transaction on Circuits and Systems for Video Technology (TCSVT) **21**(4), 381–392 (2011)
42. Perronnin, F., Liu, Y., Sánchez, J., Poirier, H.: Large-scale image retrieval with compressed fisher vectors. In: Proc. of Computer Vision and Pattern Recognition (CVPR) (2010)
43. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: Proc. of European Conference on Computer Vision (ECCV) (2010)
44. Baecchi, C., Turchini, F., Seidenari, L., Bagdanov, A.D., Bimbo, A.D.: Fisher vectors over random density forests for object recognition. In: Proc. of International Conference on Pattern Recognition (ICPR) (2014)
45. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: Delving deep into convolutional nets. arXiv preprint arXiv:1405.3531 (2014)

46. Bengio, Y.: Learning deep architectures for AI. Foundations and Trends in Machine Learning **2**(1), 1–127 (2009). DOI 10.1561/2200000006
47. Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D., Bengio, Y.: Theano: new features and speed improvements. arXiv preprint arXiv:1211.5590 (2012)
48. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. Tech. rep., CS224N Project Report, Stanford (2009)
49. Jiang, L., Yu, M., Zhou, M., Liu, X., Zhao, T.: Target-dependent Twitter sentiment classification. In: Proc. of ACL Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (HLT) (2011)
50. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology **61**(12), 2544–2558 (2010)