Twitter Sentiment Analysis via Bi-sense Emoji Embedding and Attention-based LSTM

Yuxiao Chen*
Department of Computer Science
University of Rochester
Rochester, NY
ychen211@cs.rochester.edu

Jianbo Yuan*
Department of Computer Science
University of Rochester
Rochester, NY
jyuan10@cs.rochester.edu

Quanzeng You
Microsoft Research AI
Redmond, WA
quyou@microsoft.com

Jiebo Luo
Department of Computer Science
University of Rochester
Rochester, NY
jluo@cs.rochester.edu

ABSTRACT

Sentiment analysis on large-scale social media data is important to bridge the gaps between social media contents and real world activities including political election prediction, individual and public emotional status monitoring and analysis, and so on. Although textual sentiment analysis has been well studied based on platforms such as Twitter and Instagram, analysis of the role of extensive emoji uses in sentiment analysis remains light. In this paper, we propose a novel scheme for Twitter sentiment analysis with extra attention on emojis. We first learn *bi-sense* emoji embeddings under positive and negative sentimental tweets individually, and then train a sentiment classifier by attending on these bi-sense emoji embeddings with an attention-based long short-term memory network (LSTM). Our experiments show that the bi-sense embedding is effective for extracting sentiment-aware embeddings of emojis and outperforms the state-of-the-art models. We also visualize the attentions to show that the bi-sense emoji embedding provides better guidance on the attention mechanism to obtain a more robust understanding of the semantics and sentiments.

CCS CONCEPTS

• **Information systems** → **Sentiment analysis**; • **Computing methodologies** → **Learning latent representations**;

KEYWORDS

Sentiment analysis, emoji, bi-sense embedding, attention

*Yuxiao Chen and Jianbo Yuan contributed equally to this work.

1 INTRODUCTION

The rapid growth of social media platforms such as Twitter provides rich multimedia data in large scales for various research opportunities, such as sentiment analysis which focuses on automatically sentiment (positive and negative) prediction on given contents. Sentiment analysis has been widely used in real world applications by analyzing the online user-generated data, such as election prediction, opinion mining and business-related activity analysis. Emojis, which consist of various symbols ranging from cartoon facial expressions to figures such as flags and sports, are widely used in daily communications to express people's feelings [1]. Since their first release in 2010, emojis have taken the place of emoticons (such as ":-)" and ":-P") [37] to create a new form of language for social media users [4]. According to recent science reports, there are 2,823 emojis in unicode standard in *Emoji 11.0* [2], with over 50% of the Instagram posts containing one or more emojis [11] and 92% of the online population using emojis [40].

The extensive use of emojis has drawn a growing attention from researchers [19, 25] because the emojis convey fruitful semantical and sentimental information to visually complement the textual information which is significantly useful in understanding the embedded emotional signals in texts [6]. For example, emoji embeddings have been proposed to understand the semantics behind the emojis [12, 26], and the embedding vectors can be used to visualize and predict emoji usages given their corresponding contexts. Previous work also shows that, it is useful to pre-train a deep neural network on an emoji prediction task with pre-trained emoji embeddings to learn the emotional signals of emojis for other tasks including sentiment, emotion and sarcasm prediction [15]. However, the previous literatures lack in considerations of the linguistic complexities and diversity of emoji. Therefore, previous emoji embedding methods fail to handle the situation when the semantics or sentiments of the learned emoji embeddings contradict the information from the corresponding contexts [19], or when the emojis convey multiple

[1]Real time emoji tracker: http://emojitracker.com/
[2]https://emojipedia.org/emoji-11.0/

Table 1: Tweet examples with emojis. The sentiment ground truth is given in the second column. The examples show that inconsistent sentiments exist between emojis and texts.

Emoji	Sentiment	Tweets
😁	Positive	Good morning Justin!!! I hope u have an amazing Friday :) Don't forget to smile 😁.
		That's awesome :) I'm super keen to hear/see it all 😁.
	Negative	I really hate times square personally it's too busy (I'm claustrophobic 😁).
		Not very good when your sat waiting for your food and there is a queue forming to complain to a manager 😁.
😭	Positive	This weather is perfect! 😭 It's just the change I needed.
		The dresses I ordered arrived this morning and they're so pretty 😭.
	Negative	Worst headache ever and feel so sick, mum where are you 😭.
		This nurse always comes mad early in the morning I'm mad tired 😭.

senses of semantics and sentiments such as (😂 and 🙃). In practice, emojis can either summarize and emphasis the original tune of their contexts, or express more complex semantics such as irony and sarcasm by being combined with contexts of contradictory semantics or sentiments. For the examples shown in Table 1, the emoji (😁) is of consistent sentiment with text to emphasis the sentiment, but is of the opposite sentiment (positive) to the text sentiment (negative) example 3 and 4 to deliver a sense of sarcasm. Conventional emoji analysis can only extract single embedding of each emoji, and such embeddings will confuse the following sentiment analysis model by inconsistent sentiment signals from the input texts and emojis. Moreover, we consider the emoji effect modeling different from the conventional multimodal sentiment analysis which usually includes images and texts in that, image sentiment and text sentiment are usually assumed to be consistent [45] while it carries no such assumption for texts and emojis.

To tackle such limitations, we propose a novel scheme that consists of an attention-based recurrent neural network (RNN) with robust *bi-sense* emoji embeddings. Inspired by the word sense embedding task in natural language processing (NLP) [21, 24, 39] where each sense of an ambiguous word responds to one unique embedding vector, the proposed *bi-sense* embedding is a more robust and fine-grained representation of the complicated semantics for emojis where each emoji is embedded into *two distinct vectors*, namely positive-sense and negative-sense vector, respectively. For our specific task which is Twitter sentiment analysis [23, 38], we initialize the bi-sense embedding vectors together with word embedding vectors using word embedding algorithm *fasttext* [7] by extracting two distinct embeddings for each emoji according to the sentiment of its corresponding textual contexts, namely *bi-sense* embedding. A long short-term memory (LSTM) based recurrent neural network is then used for predicting sentiments which is integrated with the pre-trained emoji embedding features by a context-guide and self-selected attention mechanism. Because most of the previous Twitter sentiment datasets exclude emojis and there exists little resource that contains sufficient emoji-tweets with sentiment labels, we construct our own emoji-tweets dataset by automatically generating weak labels using a rule-based sentiment analysis algorithm *Vader* [20] for pre-traning the networks, and manually labeling a subset of tweets for fine tuning and testing purposes.

The experimental results demonstrate that the bi-sense emoji embedding is capable of extracting more distinguished information from emojis and outperforms the state-of-the-art sentiment analysis models with the proposed attention-based LSTM networks. We further visualize the bi-sense emoji embedding to obtain the sentiments and semantics learned by the proposed approach.

The main contributions of this paper are summarized as follows.
- We propose a novel bi-sense embedding scheme that learns more robust and fine-grained representations of the complex semantic and sentiment information from emojis.
- We propose attention-based LSTM networks to encode both texts and bi-sense emoji embedding which outperform the state-of-the-art sentiment analysis models. The networks can be further extended to tackle tasks with multi-sense embedding inputs.

2 RELATED WORK

2.1 Sentiment Analysis

Sentiment analysis is to extract and quantify subjective information including the status of attitudes, emotions and opinions from a variety of contents such as texts, images and audios [47]. Sentiment analysis has been drawing great attentions because of its wide applications in business and government intelligence, political science, sociology and psychology [2, 3, 13, 33]. From a technical perspective, textual sentiment analysis is first explored by researchers as an NLP task. Methods range from lexical-based approaches using features including keywords [5, 36] where each word corresponds to a sentiment vector with entries representing the possibility of the word and each sentiment and phase-level features (n-grams and unigrams) [34, 43], to deep neural network based embedding approaches including skip-grams, continuous bag-of-words (CBoW) and skip-thoughts [7, 22, 29, 30]. It was until recent years when researchers start focusing on image and multimodal sentiments [8, 46] and analyzing how to take advantage of the cross-modality resources [44, 45]. For multimodal sentiment analysis, an underlying assumption is that both modalities express similar sentiment and such similarity is enforced in order to train a robust sentiment inference model [45]. However, the same assumption does not stand in modeling textual tweets and emojis because the complexities of natural language exist extensively, such as the use of irony, jokes, sarcasm, etc. [15].

2.2 Emojis and Sentiment Analysis

With the overwhelming development of Internet of Things (IOT), the growing accessibility and popularity of subjective contents have provided new opportunities and challenges for sentiment analysis [35]. For example, social medias such as Twitter and Instagram have been explored because the massive user-generated contents with rich user sentiments [1, 16, 34] where emojis (and emoticons) are extensively used. Non-verbal cues of sentiment, such as emoticon which is considered as the previous generation of emoji, has been studied for their sentiment effect before emojis take over [18, 27, 48]. For instance, [18, 48] pre-define sentiment labels to emoticons and construct a emoticon-sentiment dictionary. [27] applies emoticons for smoothing noisy sentiment labels. Similar work from [31] first considers emoji as a component in extracting the lexical feature for further sentiment analysis. [32] constructs an emoji sentiment ranking based on the occurrences of emojis and the human-annotated sentiments of the corresponding tweets where each emoji is assigned with a sentiment score from negative to positive [3], similar to the SentiWordNet [14]. However, the relatively intuitive use of emojis by lexical- and dictionary-based approaches lacks insightful understanding of the complexed semantics of emojis. Therefore, inspired by the success of word semantic embedding algorithms such as [7, 30], [12] obtains semantic embeddings of each emoji by averaging the words from its descriptions [4] and shows it is effective to take advantage of the emoji embedding for the task of Twitter sentiment analysis. [26] proposes a convoluntional neural network to predict the emoji occurrence and jointly learns the emoji embedding via a matching layer based on cosine similarities. Despite the growing popularity of Twitter sentiment analysis, there is a limited number of emoji datasets with sentiment labels available because previous studies usually filter out urls, emojis and sometimes emoticons. However, [15] shows that it is effective to extract sentiment information from emojis for emotion classification and sarcasm detection tasks in the absence of learning vector-based emoji representations by pre-training a deep neural network to predict the emoji occurrence.

3 METHODOLOGY

We propose two mechanisms, namely Word-guide Attention-based LSTM and Multi-level Attention-based LSTM, to take advantage of bi-sense emoji embedding for the sentiment analysis task. The frameworks of these two methods are shown in Figure 1 and Figure 2, respectively. Our workflow includes the following steps: initialization of bi-sense emoji embedding, generating senti-emoji embedding based on self-selected attention, and sentiment classification via the proposed attention-based LSTM networks.

3.1 Bi-sense Embedding

Recent research shows great success in word embedding task such as *word2vec* and *fasttext* [7, 29]. We use *fasttext* to initialize emoji embeddings by considering each emoji as a special word, together with word embeddings. The catch is, different from conventional approaches where each emoji responds to *one* embedding vector (as we call **word-emoji embedding**), we embed each emoji into
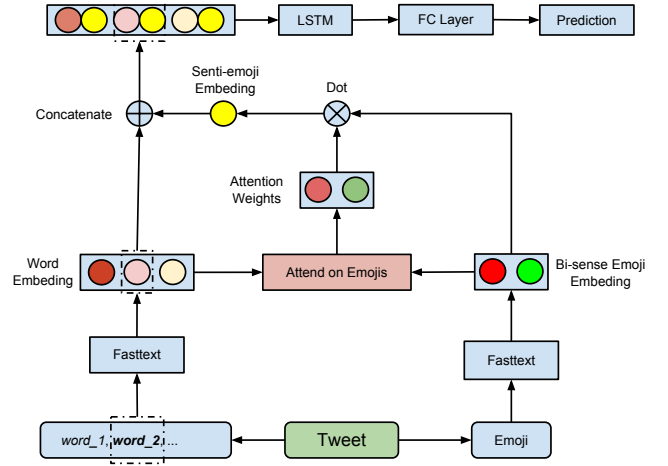

Figure 1: Sentiment analysis via bi-sense emoji embedding and attention-based LSTM network (WATT-BiE-LSTM).

two distinct vectors (**bi-sense emoji embedding**): we first assign two distinct tokens to each emoji, of which one is for the particular emoji used in positive sentimental contexts and the other one is for this emoji used in negative sentimental contexts (text sentiment initialized by *Vader* [20], details will be discussed in Section 4.1), respectively; the same *fasttext* training process is used to embed each token into a distinct vector, and we thus obtain the positive-sense and negative-sense embeddings for each emoji.

The *word2vec* is based on the *skip-gram* model whose objective is to maximize the log likelihood calculated by summing the probabilities of current word occurrences given a set of the surrounding words. The *fasttext* model is different by formatting the problem as a binary classification task to predict the occurrence of each context word, with negative samples being randomly selected from the absent context words. Given an input word sequence $\{w_1, w_2, ..., w_T\}$, and the context word set W_{c_t} and the set of negative word samples W_{n_t} of the current word w_t, the objective function is obtained based on binary logistic loss as in Equation 1:

$$\sum_{t=1}^{T} \left[\sum_{w_c \in W_{c_t}} \mathcal{L}\left(s\left(w_t, w_c\right)\right) + \sum_{w_n \in W_{n_t}} \mathcal{L}\left(s\left(w_t, w_n\right)\right) \right] \quad (1)$$

where $\mathcal{L}(s(\cdot, \cdot))$ denotes the logistic loss of the score function $s(\cdot, \cdot)$ which is computed by summing up the scalar products between the n-gram embeddings of the current word and the context word embedding which is different from *word2vec* where the score is the scalar product between the current word and the context word embedding. We select *fasttext* over *word2vec* mainly because its computational efficiency. In general, the two models yield competitive performances and the comparison between word embeddings is beyond our discussion. Therefore we only show the results derived by the *fasttext* initialization within the scope of this work.

3.2 Word-guide Attention-based LSTM

Long short-term memory (LSTM) units have been extensively used to encode textual contents. The basic encoder model consists of a text embedding layer, LSTMs layer, and fully-connected layers for further tasks such as text classifications based on the encoded feature. The operations in an LSTM unit for time step t is formulated in Equation 2:

$$
\begin{aligned}
\mathbf{i}_t &= \sigma(W_i \mathbf{x}_t + U_i \mathbf{h}_{t-1} + b_i) \\
\mathbf{f}_t &= \sigma(W_f \mathbf{x}_t + U_f \mathbf{h}_{t-1} + b_f) \\
\mathbf{o}_t &= \sigma(W_o \mathbf{x}_t + U_o \mathbf{h}_{t-1} + b_o) \\
\mathbf{g}_t &= \tanh(W_c \mathbf{x}_t + U_c \mathbf{h}_{t-1} + b_c) \\
\mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \\
\mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t)
\end{aligned}
\tag{2}
$$

where $\mathbf{h}_t$ and $\mathbf{h}_{t-1}$ represent the current and previous hidden states, $\mathbf{x}_t$ denotes the current LSTM input and here we use the embedding $\mathbf{w}_t$ of the current word w_t, and W and U denote the weight matrices [17]. In order to take advantage of the bi-sense emoji embedding, we modify the input layer into the LSTM units. We first obtain the **senti-emoji embedding** as an weighted average of the bi-sense emoji embedding based on the self-selected attention mechanism. Let $\mathbf{e}_{t,i}, i \in (1, m)$ represent the i-th sense embedding of emoji e_t ($m = 2$ in our bi-sense embedding), and $f_{att}(\cdot, \mathbf{w}_t)$ denote the attention function conditioned on the current word embedding, the attention weight α_i and senti-emoji embedding vector $\mathbf{v}_t$ is formulated as follows:

$$
\begin{aligned}
u_{t,i} &= f_{att}(\mathbf{e}_{t,i}, \mathbf{w}_t) \\
\alpha_{t,i} &= \frac{\exp(u_{t,i})}{\sum_{i=1}^{m} \exp(u_{t,i})} \\
\mathbf{v}_t &= \sum_{i=1}^{m} (\alpha_{t,i} \cdot \mathbf{e}_{t,i})
\end{aligned}
\tag{3}
$$

We choose a fully-connected layer with *ReLU* activation as the attention function, and the attention vector v_t is concatenated with the word embedding as the new input of the LSTM. Thus the input vector $\mathbf{x}_t$ in Equation 2 becomes $[\mathbf{w}_t, \mathbf{v}_t]$. The output of the final LSTM unit is then fed into a fully-connected layer with *sigmoid* activation to output the tweet sentiment and binary cross-entropy loss is used as the objection function (Equation 4) where N is the total number of samples. The motivation behind this model is that each context word guides the attention weights in order to enforce the model to self-select which embedding sense it should attend on. Therefore we denote this model as the Word-guide Attention-based LSTM with Bi-sense emoji embedding (**WATT-BiE-LSTM**).

$$
\mathcal{L}(\theta) = -\frac{1}{N} \sum_{i=1}^{N} (y_i \log(p_i) + (1 - y_i) \log(1 - p_i))
\tag{4}
$$

3.3 Multi-level Attention-based LSTM

There is another way of formulating the attention mechanism where the attention weights indicate how the image information (which is emoji in our case) is distributed through the context words as proposed in [9, 42]. The modified senti-emoji embedding vector $\mathbf{v}$ is thus at the tweet-level instead of the word-level in Equation 3 by
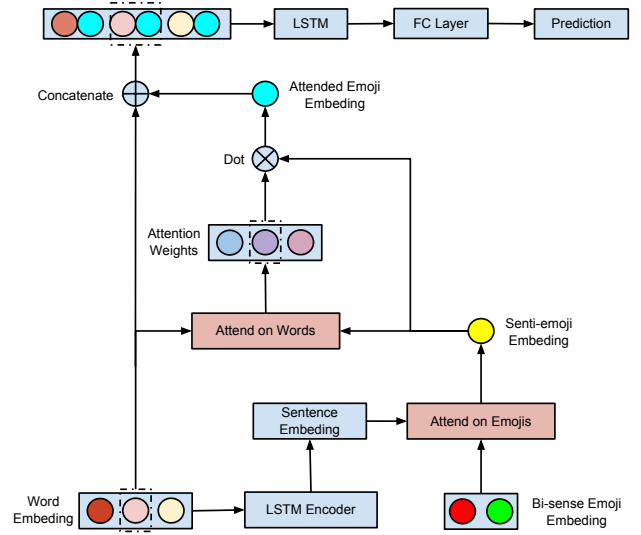

Figure 2: Multi-level attention-based LSTM with bi-sense emoji embedding (MATT-BiE-LSTM).

replacing the $\mathbf{w}_t$ with the final state vector $\mathbf{h}$ outputted from the last LSTM unit, as shown in Equation 5:

$$
\begin{aligned}
\alpha_i &= \frac{\exp(f_{att}(\mathbf{e}_i, \mathbf{h}))}{\sum_{i=1}^{m} \exp(f_{att}(\mathbf{e}_i, \mathbf{h}))} \\
\mathbf{v}' &= \sum_{i=1}^{m} (\alpha_i \cdot \mathbf{e}_i)
\end{aligned}
\tag{5}
$$

The derived senti-emoji embedding $\mathbf{v}'$ is then used to calculate an additional layer of attention following [9, 42]. Given the input tweet sequence $\{w_1, w_2, ..., w_T\}$, the attention weight $\alpha'_t, t \in (1, T)$ conditioned on the senti-emoji embedding is formulated as follows:

$$
\alpha'_t = \frac{\exp(f_{att}(\mathbf{w}_t, \mathbf{v}'))}{\sum_{i=1}^{m} \exp(f_{att}(\mathbf{w}_t, \mathbf{v}'))}
\tag{6}
$$

Therefore, we construct the new input u_t to each LSTM unit by concatenating the original word embedding and the attention vector in Equation 7 to distribute the senti-emoji information to each step. This model is called Multi-level Attention-based LSTM with Bi-sense Emoji Embedding (**MATT-BiE-LSTM**). We choose the same binary cross-entropy as the loss function with the same network configuration with WATT-BiE-LSTM.

$$
\mathbf{u} = [\mathbf{w}_t, \alpha'_t \cdot \mathbf{v}']
\tag{7}
$$

4 EXPERIMENTS

4.1 Data Collection and Annotation

Data Collection

We construct our own Twitter sentiment dataset by crawling tweets through the REST API [5] which consists of 350,000 users and is magnitude larger comparing to previous work. We collect up to 3,200

---

[5]https://developer.twitter.com/en/docs

Table 2: Top-10 Most Frequently Used Emojis.

Ranking	Emoji	AA-Sentiment			HA-Sentiment		
		Positive	Negative	Pos-Ratio	Positive	Negative	Pos-Ratio
1	😂	164,677	62,816	0.724	333	162	0.673
2	❤️	146,498	4,715	0.969	184	32	0.852
3	😊	105,329	4,566	0.958	181	23	0.887
4	😍	66,905	2,678	0.962	194	7	0.965
5	😘	62,369	2,155	0.967	93	7	0.930
6	😌	53,913	2,079	0.963	101	13	0.886
7	😭	31,077	24,519	0.559	56	177	0.240
8	😉	42,543	4,212	0.910	128	19	0.871
9	👍	42,919	3,182	0.931	91	25	0.784
10	😁	38,316	4,514	0.895	112	24	0.824

tweets from each user and follow the standard tweet preprocessing procedures to remove the tweets without emojis and tweets containing less than ten words, and contents including the urls, mentions, and emails.

Data Annotation

For acquiring the sentiment annotations, we first use *Vader* which is a rule-based sentiment analysis algorithm [20] for text tweets only to generate weak sentiment labels. The algorithm outputs sentiment scores ranging from -1 (negative) to 1 (positive) with neutral in the middle. We consider the sentiment analysis as a binary classification problem (positive sentiment and negative sentiment), we filter out samples with weak prediction scores within $(-0.60, 0.60)$ and keep the tweets with strong sentiment signals. Emoji occurrences are calculated separately for positive tweets and negative tweets, and threshold is set to 2,000 to further filter out emojis which are less frequently used in at least one type of sentimental text. In the end, we have constructed a dataset with 1,492,065 tweets and 55 frequently used emojis in total. For the tweets with an absolute sentiment score over 0.70, we keep the auto-generated sentiment label as ground truth because the automatic annotation is reliable with high sentiment scores. On the other hand, we select a subset of the tweets with absolute sentiment scores between $(0.60, 0.70)$ for manual labeling by randomly sampling, following the distribution of emoji occurrences where each tweet is labeled by two graduate students. Tweets are discarded if the two annotations disagree with each other or they are labeled as neutral. In the end, we have obtained 4,183 manually labeled tweets among which 60% are used for fine-tuning and 40% are used for testing purposes. The remainder of the tweets with automatic annotations are divided into three sets: 60% are used for pre-training the bi-sense and conventional emoji embedding, 10% for validation and 30% are for testing. We do not include a "neutral" class because it is difficult to obtain valid neutral samples. For auto-generated labels, the neutrals are the samples with low absolute confidence scores and their sentiments are more likely to be model failures other than "true neutrals". Moreover, based on the human annotations, most of the tweets with emojis convey non-neutral sentiment and only few neutral samples are observed during the manual labeling which are excluded from the manually labeled subset.

In order to valid our motivation that emojis are also extensively used in tweets that contain contradictory information to the emoji sentiments, we calculate the emoji usage in Table 2 according to the sentiment labels where *Pos-Ratio* means the percentage of each emoji occurs in the positive tweets over its total number of occurrences, *AA* and *HA* indicate automatic-annotation and human-annotation, respectively. We present the top-10 most frequently used emojis in our dataset and observe a slight difference in the Pos-Ratios between AA and HA dataset because of the randomness involved in the sampling process. Results from both of the datasets show a fair amount of emoji use in both positive and negative tweets. For example, it is interesting to notice that emoji (😭) occurs more in the positive tweets in with the automatic annotations, while emojis with strong positive sentiment have also been used in negative tweets with about 5% occurrences, such as (❤️, 😊, and 😍). Given the averaged positive ratio among all emojis in the whole dataset is about 74% and that most emojis have been extensively used in tweets containing both positive and negative sentiments, it suggests that distinguishing the emoji occurrences in both sentiments via bi-sense embedding is worth investigating. Additionally, we observe the *Pos-Ratios* of the AA-sentiment and HA-sentiment have little differences which are due to two main reasons: 1) Some tweets we sampled to construct the HA-sentiment are discarded because the annotators have disagreements and we only keep the samples that we are confident about; 2) Tweets with absolute sentiment scores between (0.60,0.70) are selected for manual labeling as discussed in Section 4.1, which are lower than the tweets used to construct the AA-sentiment (0.7 and above). The lower sentiment scores indicate that Vader is less reliable on the samples of HA-sentiment dataset and the sentiments of these tweets are more likely to be affected by emojis.

4.2 Sentiment Analysis

Models

We set up the baselines and proposed models as follows:
LSTM with text embedding: CNNs and LSTMs are widely used to encode textual contents for sentiment analysis in [10, 41] and many online tutorials. Here we select the standard LSTM with pre-trained

Table 3: Twitter Sentiment Analysis.

Models	AA-Sentiment					HA-Sentiment				
	Precision	Recall	ROC Area	Accuracy	F1 Score	Precision	Recall	ROC Area	Accuracy	F1 Score
T-LSTM	0.921	**0.901**	0.931	0.866	0.911	0.708	0.825	0.774	0.707	0.762
E-LSTM	0.944	0.86	0.933	0.855	0.900	0.816	0.825	0.855	0.794	0.820
ATT-E-LSTM	0.948	0.890	0.954	0.879	0.918	0.825	0.868	0.878	0.820	0.846
BiE-LSTM	0.961	0.891	0.966	0.890	0.924	0.822	0.881	0.898	0.824	0.850
MATT-BiE-LSTM	**0.972**	0.895	**0.975**	**0.900**	**0.932**	**0.831**	0.872	0.890	0.826	0.851
WATT-BiE-LSTM	0.949	0.895	0.960	0.883	0.921	0.830	**0.889**	**0.899**	**0.834**	**0.859**

word embedding as input, and add one fully-connected layer with *sigmoid* activation top of the LSTM encoder (same as all other models), denoted as **T-LSTM**.

LSTM with emoji embedding: We consider the emoji as one special word and input both pre-trained text and emoji embeddings into the same LSTM network, namely **E-LSTM**. Similarly, we concatenate the pre-trained bi-sense emoji embedding as one special word to feed into the LSTM network. This model is called **BiE-LSTM**.

Attention-based LSTM with emojis:We also use the word-emoji embedding to calculate the emoji-word attention following Equation 6 and 7, and the only difference is that we replace the attention-derived senti-emoji embedding with the pre-trained word-emoji embedding by *fasttext*, denoted as **ATT-E-LSTM**.

LSTM with bi-sense emoji embedding (proposed): As we have introduced in Section 3.2, we propose two attention-based LSTM networks based on bi-sense emoji embedding, denoted as **MATT-BiE-LSTM** and **WATT-BiE-LSTM**.

Evaluation

We evaluate the baseline and proposed models on sentiment analysis by F1 scores and accuracies based on the auto-annotated testing set (AA-Sentiment) and human-annotated testing set (HA-Sentiment), as shown in Table 3. We only test the models after fine-tuning with a subset of the samples with human annotations because training exclusively on the samples with auto-generated weak labels results in relatively poor performances when tested with human annotated data indicating the models after fine-tuning are more robust. The F1 scores and accuracies are overall higher with the AA-Sentiment than the results with HA-sentiment, indicating that the HA-Sentiment is a more challenging task and the sentiments involved are more difficult to identify supported by their relatively lower sentiment scores returned from *Vader*. We still, however, observe competitive results from HA-Sentiment showing that the models are well-trained and robust to noisy labels with the help of fine-tuning with human annotated data. The T-LSTM baseline achieves decent performance in both experiments with accuracies of 86.6% and 70.7% showing that LSTM is an effective encoder for sentiment analysis as suggested by the references. The models with proposed bi-sense emoji embedding obtain accuracies over 82.4% and we observe improvements on the performance with the attention-based LSTM from our proposed model MATT-BiE-LSTM and WATT-BiE-LSTM, which is consistent with that ATT-E-LSTM (F1@84.6%, accuracy@82.0% on HA-Sentiment) outperforms significantly T-LSTM and E-LSTM.

Emoji information is useful in sentiment analysis. Most models outperforms the baseline T-LSTM in both dataset suggesting that the emoji information is useful for sentiment analysis as a complement to the textual contents, even with the naive use of emoji embeddings (E-LSTM) when tested with HA-Sentiment. We observe that E-LSTM obtains similar performance to T-LSTM with AA-Sentiment but a significant gain over the T-LSTM when tested with HA-Sentiment indicating that sentiment information is helpful and necessary when the hidden sentiment is relatively subtle and the task is more challenging.

Bi-sense emoji embedding helps. All the models using bi-sense emoji embedding perform significantly better than the baseline models without emoji feature or with word-emoji embedding. BiE-LSTM outperforms T-LSTM and E-LSTM significantly with the same utilization of emoji embedding indicates that the proposed bi-sense emoji embedding is capable of extracting more informative and distinguishable vectors over the use of conventional word embedding algorithms, which is consistent based on the comparisons between the proposed models (MATT-BiE-LSTM and WATT-BiE-LSTM) with bi-sense emoji embedding and the baseline model ATT-E-LSTM with word-emoji embedding and attention.

Attention mechanism aligns and performs well with bi-sense embedding. MATT-BiE-LSTM and WATT-BiE-LSTM obtain similar performances when tested on both *Vader* and human annotated samples, though their ways of computing the attention (weights and vectors) are different that WATT computes attention weights and the senti-emoji embeddings guided by each word, and MATT obtains the senti-emoji embedding based on the LSTM encoder on the whole contexts and computes the attention weights of the senti-emoji embedding across all words. Both models outperforms the state-of-the-art baseline models including ATT-E-LSTM. The proposed attention-based LSTM can be further extended to handle tasks involving multi-sense embedding as inputs, such as the word-sense embedding in NLP, by using context-guide attention to self-select how much to attend on each sense of the embeddings each of which correspond to a distinct sense of semantics or sentiments. In this way we are able to take advantage of the more robust and fine-grained embeddings.

4.3 Qualitative Analysis

In order to obtain insights about why the more fine-grained bi-sense emoji embedding helps in understanding the complexed sentiments behind tweets, we visualize the attention weights for ATT-E-LSTM
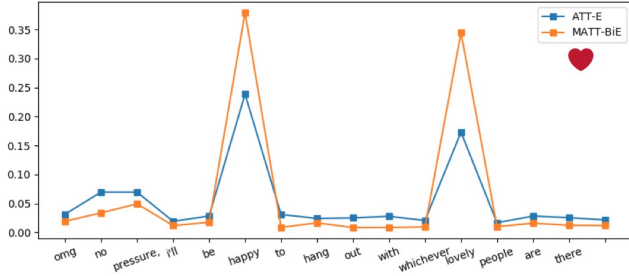
(a) OMG no pressure, I'll be happy to hang out with whichever lovely people are there.
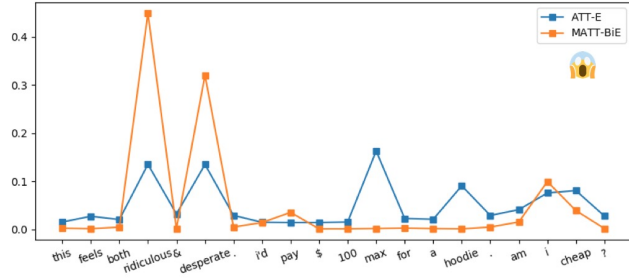
(b) This feels both ridiculous & desperate. I'd pay $100 max for a hoodie. Am I cheap?
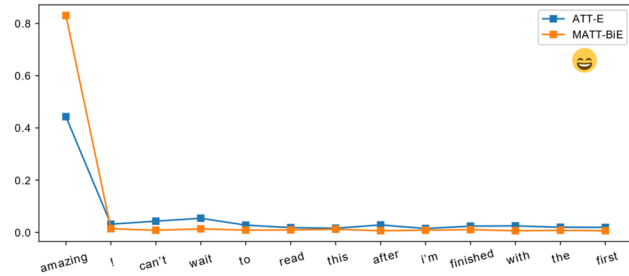
(c) Amazing! Can't wait to read this after I'm finished with the first.

Figure 3: Attention weights obtained by senti-emoji embedding and word-emoji embedding across words. Tweet contexts are given in sub-captions.

and MATT-BiE-LSTM for comparison. The example tweets with corresponding attention weights calculated by word-emoji embedding and senti-emoji embedding are shown in Figure 3, where the contexts are presented in the captions. The emojis used are ❤️, 😱, and 😊, respectively.

In Figure 3(a), the ATT-E-LSTM model (baseline) assigns relatively more weights on the word "no" and "pressure", while MATT-BiE-LSTM attends mostly on the word "happy" and "lovely". The different attention distributions suggest that the proposed senti-emoji embedding is capable of recognizing words with strong sentiments that are closely related to the true sentiment even with the presence of words with conflicting sentiments, such as "pressure" and "happy". while ATT-E-LSTM tends to pick up all sentimental

words which could raise confusions. The senti-emoji embedding is capable of extracting representations of complexed semantics and sentiments which help guide the attentions even in cases when the word sentiment and emoji sentiment are somewhat contradictory to each other. From Figure 3(b) and (c) we can observe that the ATT-E-LSTM assigns more weights on the sentiment-irrelevant words than the MATT-BiE-LSTM such as "hoodies", "wait" and "after", indicating that the proposed model is more robust to irrelevant words and concentrates better on important words. Because of the senti-emoji embedding obtained through bi-sense emoji embedding and the sentence-level LSTM encoding on the text input (described in Section 3.2), we are able to construct a more robust embedding based on the semantic and sentiment information from the whole context compared to the word-emoji embedding used in ATT-E-LSTM which takes only word-level information into account.

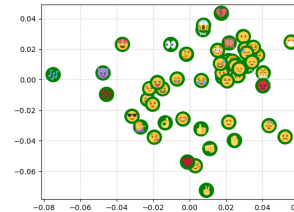## 4.4 Bi-sense Emoji Embedding Visualization

To gain further insights on the bi-sense emoji embedding, we use *t-SNE* [28] to project high-dimensional bi-sense embedding vectors into a two-dimensional space and preserving the relative distances between the embedding vectors at the same time. In Figure 4 we visualize the bi-sense emoji embedding, positive-sense embedding, negative-sense embedding and the subtraction between positive and negative sense embeddings of each emoji, respectively. The subtraction of an emoji between its two sense embeddings indicates the semantic differences between emoji usages in positive and negative sentimental contexts, similarly to the objective of word embeddings
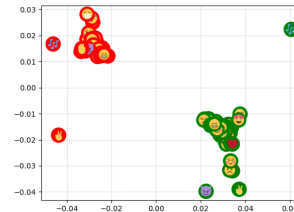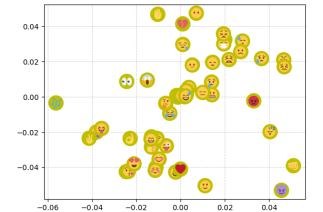

(a) Bi-sense Emoji Embedding
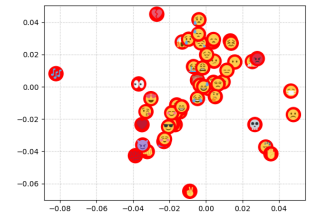
(b) Positive-sense Embedding

(c) Negative-sense Embedding

(d) Positive-sense − Negative-sense

Figure 4: *t-SNE* visualization of bi-sense emoji embedding. Positive-sense embeddings are paired with red circles, negative-sense embeddings are paired with green circles, and their subtractions are paired with yellow circles, respectively. Best viewed when zoomed in.

[30]. The positive-sense of emoji (✌️ and 🎶), and the negative-sense of emoji (✌️, 🎶 and 😈) are embedded far from the two main clusters as observed in Figure 4(a), suggesting that the semantics of these emojis are different from the other popular emojis. The positive-sense embedding and negative-sense embeddings are clustered well with no intersection with each other. Such observation supports our objective of applying bi-sense emoji embedding because there exist such significant differences in the semantics of each emoji when appears in positive and negative sentimental contexts, and it is well-motivated to consider the emoji usages individually according to the sentiment of the contexts to extract the more fine-grained bi-sense embedding. Additionally, we observe consistent patterns in the Figure 4(b), (c) and (d) where the sentiments conveyed in the emojis become an important factor. For example, emojis with positive sentiments such as (😄, 😋 and 😊), and emojis with negative sentiment such as (😥, 😭 and 😔) are embedded into one clusters in both positive-sense and negative-sense space. The embedding subtractions of emojis in Figure 4(d) shows the different usages of emojis across sentiments are similar between emojis and preserve the cluster patterns observed in Figure 4 (b) and (c).

5 CONCLUSIONS

In this paper, we present a novel approach to the task of sentiment analysis and achieve the state-of-the-art performance. Different from the previous work, our method combines a more robust and fine-grained bi-sense emoji embedding that effectively represents complex semantic and sentiment information, with attention-based LSTM networks that selectively attend on the correlated sense of the emoji embeddings, and seamlessly fuse the obtained senti-emoji embeddings with the word embeddings for a better understanding of the rich semantics and sentiments involved. In the future, we plan to further extend our attention-based LSTM with bi-embedding work frame to tackle tasks involving multi-sense embedding such as the learning and applications of word-sense embedding.

ACKNOWLEDGEMENT

REFERENCES

[1] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media*. Association for Computational Linguistics, 30–38.
[2] Hanaa A. Aldahawi. 2015. *Mining and analysing social network in the oil business: Twitter sentiment analysis and prediction approaches*. Ph.D. Dissertation. Cardiff University, UK.
[3] Ghaith Abdulsattar A. Jabbar Alkubaisi, Siti Sakira Kamaruddin, and Husniza Husni. 2018. Stock Market Classification Model Using Sentiment Analysis on Twitter Based on Hybrid Naive Bayes Classifiers. *Computer and Information Science* 11, 1 (2018), 52–64.
[4] Hamza Alshenqeeti. 2016. Are emojis creating a new or old visual language for new generations? A socio-semiotic study. *Advances in Language and Literary Studies* 7, 6 (2016), 56–69.
[5] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta.*
[6] Francesco Barbieri, Germán Kruszewski, Francesco Ronzano, and Horacio Saggion. 2016. How Cosmopolitan Are Emojis?: Exploring Emojis Usage and Meaning over Different Languages with Distributional Semantics. In *Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016.* 531–535.

[7] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *TACL* 5 (2017), 135–146.
[8] Damian Borth, Tao Chen, Rongrong Ji, and Shih-Fu Chang. 2013. Sentibank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content. In *Proceedings of the 21st ACM international conference on Multimedia.* ACM, 459–460.
[9] Tianlang Chen, Yuxiao Chen, Han Guo, and Jiebo Luo. 2018. When E-commerce Meets Social Media: Identifying Business on WeChat Moment Using Bilateral-Attention LSTM. In *Proceedings of the 27th International Conference on World Wide Web Companion, Lyon, France, April, 2018.*
[10] Mathieu Cliche. 2017. BB_twtr at SemEval-2017 Task 4: Twitter Sentiment Analysis with CNNs and LSTMs. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017.* 573–580.
[11] Thomas Dimson. 2015. Emojineering part 1: Machine learning for emoji trends. *Instagram Engineering Blog* 30 (2015).
[12] Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bosnjak, and Sebastian Riedel. 2016. emoji2vec: Learning Emoji Representations from their Description. In *Proceedings of The Fourth International Workshop on Natural Language Processing for Social Media, SocialNLP@EMNLP 2016, Austin, TX, USA, November 1, 2016.* 48–54.
[13] Tarek Elghazaly, Amal Mahmoud, and Hesham A. Hefny. 2016. Political Sentiment Analysis Using Twitter Data. In *Proceedings of the International Conference on Internet of Things and Cloud Computing, Cambridge, UK, March 22-23, 2016.* 11:1–11:5.
[14] Andrea Esuli and Fabrizio Sebastiani. 2006. SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006, Genoa, Italy, May 22-28, 2006.* 417–422.
[15] Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017.* 1615–1625.
[16] Roberto González-Ibánez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in Twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2.* Association for Computational Linguistics, 581–586.
[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[18] Alexander Hogenboom, Daniella Bal, Flavius Frasincar, Malissa Bal, Franciska de Jong, and Uzay Kaymak. 2013. Exploiting emoticons in sentiment analysis. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing, SAC '13, Coimbra, Portugal, March 18-22, 2013.* 703–710.
[19] Tianran Hu, Han Guo, Hao Sun, Thuy-vy Thi Nguyen, and Jiebo Luo. 2017. Spice Up Your Chat: The Intentions and Sentiment Effects of Using Emojis. In *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017, Montréal, Québec, Canada, May 15-18, 2017.* 102–111.
[20] Clayton J. Hutto and Eric Gilbert. 2014. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014.*
[21] Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2015. SensEmbed: Learning Sense Embeddings for Word and Relational Similarity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers.* 95–105.
[22] Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems.* 3294–3302.
[23] Efthymios Kouloumpis, Theresa Wilson, and Johanna D. Moore. 2011. Twitter Sentiment Analysis: The Good the Bad and the OMG!. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011.*
[24] Jiwei Li and Dan Jurafsky. 2015. Do Multi-Sense Embeddings Improve Natural Language Understanding?. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015.* 1722–1732.
[25] Weijian Li, Yuxiao Chen, Tianran Hu, and Jiebo Luo. 2018. Mining the Relationship between Emoji Usage Patterns and Personality. In *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM 2018, Stanford, California, USA, June 25-28, 2018.* 648–651.
[26] Xiang Li, Rui Yan, and Ming Zhang. 2017. Joint Emoji Classification and Embedding Learning. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data.* Springer, 48–63.
[27] Kun-Lin Liu, Wu-Jun Li, and Minyi Guo. 2012. Emoticon Smoothed Language Models for Twitter Sentiment Analysis. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*

[28] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, Nov (2008), 2579–2605.

[29] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013).

[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems.* 3111–3119.

[31] Petra Kralj Novak, Jasmina Smailović, Borut Sluban, and Igor Mozetič. 2015. Sentiment of emojis. *PloS one* 10, 12 (2015), e0144296.

[32] Petra Kralj Novak, Jasmina Smailovic, Borut Sluban, and Igor Mozetic. 2015. Sentiment of Emojis. *CoRR* abs/1509.07761 (2015). arXiv:1509.07761 http://arxiv.org/abs/1509.07761

[33] Nazan Öztürk and Serkan Ayvaz. 2018. Sentiment analysis on Twitter: A text mining approach to the Syrian refugee crisis. *Telematics and Informatics* 35, 1 (2018), 136–147.

[34] Alexander Pak and Patrick Paroubek. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta.*

[35] Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval* 2, 1–2 (2008), 1–135.

[36] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10.* Association for Computational Linguistics, 79–86.

[37] Umashanthi Pavalanathan and Jacob Eisenstein. 2015. Emoticons vs. Emojis on Twitter: A Causal Inference Approach. *CoRR* abs/1510.08480 (2015).

[38] Aliza Sarlan, Chayanit Nadam, and Shuib Basri. 2014. Twitter sentiment analysis. In *Information Technology and Multimedia (ICIMU), 2014 International Conference on.* IEEE, 212–216.

[39] Linfeng Song, Zhiguo Wang, Haitao Mi, and Daniel Gildea. 2016. Sense Embedding Learning for Word Sense Induction. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, *SEM@ACL 2016, Berlin, Germany, 11-12 August 2016.*

[40] Emoji Research Team. 2016. 2016 Emoji Report. (2016). http://cdn.emogi.com/docs/reports/2016_emoji_report.pdf.

[41] Quan-Hoang Vo, Huy-Tien Nguyen, Bac Le, and Minh-Le Nguyen. 2017. Multi-channel LSTM-CNN model for Vietnamese sentiment analysis. In *Knowledge and Systems Engineering (KSE), 2017 9th International Conference on.* IEEE, 24–29.

[42] Yequan Wang, Minlie Huang, Li Zhao, et al. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing.* 606–615.

[43] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing.* Association for Computational Linguistics, 347–354.

[44] Quanzeng You. 2016. Sentiment and emotion analysis for social multimedia: methodologies and applications. In *Proceedings of the 2016 ACM on Multimedia Conference.* ACM, 1445–1449.

[45] Quanzeng You, Liangliang Cao, Hailin Jin, and Jiebo Luo. 2016. Robust Visual-Textual Sentiment Analysis: When Attention meets Tree-structured Recursive Neural Networks. In *Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016.* 1008–1017.

[46] Jianbo Yuan, Sean Mcdonough, Quanzeng You, and Jiebo Luo. 2013. Sentribute: image sentiment analysis from a mid-level perspective. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining.* ACM, 10.

[47] Jianbo Yuan, Quanzeng You, and Jiebo Luo. 2015. Sentiment Analysis Using Social Multimedia. In *Multimedia Data Mining and Analytics - Disruptive Innovation.* 31–59. https://doi.org/10.1007/978-3-319-14998-1_2

[48] Jichang Zhao, Li Dong, Junjie Wu, and Ke Xu. 2012. MoodLens: an emoticon-based sentiment analysis system for chinese tweets. In *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012.* 1528–1531.