Information Processing and Management 56 (2019) 1633–1644

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Irony detection via sentiment-based transfer learning

Shiwei Zhang[a], Xiuzhen Zhang*[,a], Jeffrey Chan[a], Paolo Rosso[b]

[a] *RMIT University, Australia*
[b] *Universitat Politècnica de València, Spain*

ARTICLE INFO

Keywords:
Irony detection
Transfer learning

ABSTRACT

Irony as a literary technique is widely used in online texts such as Twitter posts. Accurate irony detection is crucial for tasks such as effective sentiment analysis. A text's ironic intent is defined by its context incongruity. For example in the phrase "I love being ignored", the irony is defined by the incongruity between the positive word "love" and the negative context of "being ignored". Existing studies mostly formulate irony detection as a standard supervised learning text categorization task, relying on explicit expressions for detecting context incongruity. In this paper we formulate irony detection instead as a transfer learning task where supervised learning on irony labeled text is enriched with knowledge transferred from external sentiment analysis resources. Importantly, we focus on identifying the hidden, implicit incongruity without relying on explicit incongruity expressions, as in "I like to think of myself as a broken down Justin Bieber – my philosophy professor." We propose three transfer learning-based approaches to using sentiment knowledge to improve the attention mechanism of recurrent neural models for capturing hidden patterns for incongruity. Our main findings are: (1) Using sentiment knowledge from external resources is a very effective approach to improving irony detection; (2) For detecting implicit incongruity, transferring deep sentiment features seems to be the most effective way. Experiments show that our proposed models outperform state-of-the-art neural models for irony detection.

1. Introduction

User-generated texts on social media platforms like Twitter and Facebook often involve the widespread use of creative and figurative languages like irony and sarcasm. A text utterance is perceived to be ironic if its intended meaning is opposite to what it literally expresses. The terms irony and sarcasm are often used interchangeably, despite their subtle differences in meaning (Hernańdez, Patti, & Rosso, 2016). Accurate irony detection is important for social media analysis. For example, failing to detect irony can lead to low performance for sentiment analysis, since the presence of irony often causes polarity reversal (Hernańdez & Rosso, 2016). Also, irony detection is essential for security services to discriminate potential threats from just ironic comments (Rosso et al., 2018). In contrast to most text classification tasks, irony detection is a challenging task (González-Ibáñez, Muresan, & Wacholder, 2011) that requires inferring the hidden, ironic intent, which can not be achieved by literal syntactic or semantic analysis of the textual contents. Indeed the challenge of irony detection is clearly shown in the sentiment polarity classification task of Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (Evalia 2016) (Barbieri et al., 2016). Three independent sub-tasks are included, namely subjectivity classification, polarity classification and irony detection. The

* Corresponding author.
E-mail addresses: shiwei.zhang@rmit.edu.au (S. Zhang), xiuzhen.zhang@rmit.edu.au (X. Zhang), jeffrey.chan@rmit.edu.au (J. Chan), prosso@dsic.upv.es (P. Rosso).

https://doi.org/10.1016/j.ipm.2019.04.006
Received 6 September 2018; Received in revised form 18 April 2019; Accepted 19 April 2019
Available online 16 May 2019

performance for both subjective classification and polarity classification is over 10% higher than that of irony detection in terms of the *F* Measure.

According to linguistics research, irony is the incongruity expressed between the context and statement conveyed in a piece of text (Gerrig & Goldvarg, 2000; Ivanko & Pexman, 2003; Joshi, Sharma, & Bhattacharyya, 2015). Sentiment polarity contrast is a commonly seen form of irony on Twitter (Joshi et al., 2015; Riloff et al., 2013). For example in the tweet “I love when I wake up grumpy”, “love” expresses positive polarity whereas the phrase “wake up grumpy” expresses negative polarity. The hidden sentiment polarity contrast signifies the ironic intent of the tweet. Moreover, the extent of irony perception depends on the strength of the context (“wake up grumpy”) and the strength of the statement (“love”). Explicit incongruity refers to contrast from explicit sentiment words as in “I love being ignored”, where “love” is positive and “ignore” is negative. Implicit incongruity refers to contrast from phrases expressing implicit sentiment polarity but not using explicit sentiment words, as in “I love this paper so much that I put in it my drawer”; the phrase “put in my drawer” implies a negative polarity and forms a contrast with the positive sentiment in “love”.

The task of irony detection is to classify a piece of text as ironic or non-ironic. Existing studies mostly formulate irony detection as a standard supervised learning text categorization problem. Approaches to irony detection on Twitter can be roughly classified into three classes, namely rule-based approaches, classical feature-based machine learning methods and deep neural network models. In the literature, rule-based and classical feature-based machine learning models are proposed for irony detection (See Joshi, Bhattacharyya, & Carman, 2017 and Wallace, 2015 for surveys). Recently deep learning models are applied for irony detection (Ghosh & Veale, 2017; Ghosh, Fabbri, & Muresan, 2017; Huang, Huang, & Chen, 2017; Joshi, Tripathi, Patel, Bhattacharyya, & Carman, 2016; Oraby, Harrison, Misra, Riloff, & Walker, 2017; Poria, Cambria, Hazarika, & Vij, 2016) and show better performance than classical feature-based machine learning models. Among all of the neural network-based models, attention-based models are most effective. Apart from standard attention models, a recent work (Tay, Tuan, Hui, & Su, 2018) proposed an intra-attention mechanism for sarcasm detection. Their model is looking into intricate similarities between each word pair.

Most previous studies do not study context incongruity for irony detection. A few studies focus on identifying context incongruity for irony detection, but with limitations. One previous study (Riloff et al., 2013) made use of the pattern “positive sentiment followed by negative situation” to detect irony on Twitter. The approach can miss many forms of context incongruity that do not follow this pattern. Another previous study (Joshi et al., 2015) manually engineered explicit and implicit context incongruity features for irony detection and still can capture limited context incongruity.

In this paper we formulate irony detection as a transfer learning task where supervised learning on irony labels is enriched with knowledge transferred from external sentiment analysis resources. Moreover, we focus on the key issue for irony detection – identifying the hidden, implicit incongruity without explicit incongruity expressions as well as the explicit incongruity. Our key idea is to transfer external sentiment knowledge from sentiment resources to train the deep neural model for irony detection. Resources for sentiment analysis are readily available, including sentiment lexica (Wilson, Wiebe, & Hoffmann, 2005) and sentiment corpora (Novak, Smailović, Sluban, & Mozetič, 2015).

We propose three sentiment-based transfer learning models to improve the attentive recurrent neural model for identifying explicit and implicit context incongruity for irony detection on Twitter. The three models are designed to transfer different types of sentiment knowledge. The first two methods are focused on transferring hard sentiment attention generated from a pre-defined sentiment corpus, but the hard attention in the first model is treated as an external feature while it is treated as an extra supervision signal in the second model. The last model is focused on transferring deep features from the sentiment analysis on Twitter for the irony detection task, where features from both tasks are mapped into a common latent feature space. By comparing these different approaches one can find the most effective way of using sentiment-based transfer learning for irony detection.

Main contributions of this paper are:

- We proposed three novel methods for attention-based models to incorporate sentiment features instead of feature-vector concatenation. Experiments show that the proposed methods are effective to improve the accuracy of attention models for irony detection.
- Learning deep features on sentiment tweets corpora and transferring them into the attention-based neural model is the most effective way to detect both explicit and implicit context incongruity.
- To our best knowledge, for the first time, we contrast the human-labeled and hashtag-labeled datasets for evaluation of irony detection models. We find that the human-labeled dataset is much more challenging than the hashtag-labeled dataset and gives a more accurate estimation of the performance for irony detection models in real applications. We also discussed several possible reasons that made the hashtag-labeled datasets easier for irony detection.

The rest of this paper is organized as follows: In Section 2, we state our research objective. In Section 3, we discuss the related work including both conventional methods and deep learning methods on irony detection. In Section 4, we discuss the shortcomings of using attention-based Bi-LSTM on irony detection. In Section 5, we present the details of our proposed approaches. In Section 6, we describe the experimental setup and discuss experimental results, and interpret results with attention-based visualization. In Section 7, we conclude our work.

2. Research objective

Identifying context incongruity is the key to detect the ironic intent of Twitter posts. But in the literature, automatic sarcasm/irony detection is commonly formulated as a supervised learning classification task. Given a collection of tweets annotated as either *ironic* or *non-ironic*, a classification model is trained on the annotated tweets' collection and is then applied to predict the label of unseen tweets. For instance, most previous works do not extract patterns of the context incongruity or provide clear reasoning on detecting the context incongruity, especially when using deep learning models.

Although the irony intent is mainly expressed by incongruous sentiment between the context and the statement, the limited annotated resource is a barrier for a model to fully detect those sentiment patterns given the extremely various sentiment patterns available in human languages. On the other hand, sentiment resources are widely and readily available, which could be leveraged for irony detection. We formulate irony detection as a transfer learning task where supervised learning on irony labels is enriched with knowledge transferred from external sentiment analysis resources. Specifically, our research objective is to address the following two research questions:

- How to transfer different types of sentiment knowledge for irony detection?
- How to effectively use the transferred knowledge to detect the context incongruity, especially the implicit context incongruity?

3. Related work

Approaches to irony detection on Twitter can be roughly classified into three classes, namely rule-based approaches, classical feature-based machine learning methods and deep neural network models. Rule-based approaches generally rely on linguistic features such as sentiment lexicon or hashtags to detect irony on Twitter (González-Ibáñez et al., 2011; Maynard & Greenwood, 2014; Sulis, Farías, Rosso, Patti, & Ruffo, 2016). Twitter uses hashtags to invert the literal sentiment in tweets (Maynard & Greenwood, 2014). The most popular hashtags for indicating irony include *#irony, #sarcasm* and *#not* (Sulis et al., 2016). The use of hashtags like "#sarcasm", is believed to be a replacement of linguistic markers such as exclamations and intensifiers (Kunneman, Liebrecht, Mulken, & Bosch, 2015). Classical feature-based machine learning approaches use hand-crafted features (Hernańdez et al., 2016) for irony detection, such as sentiment lexicon, subjectivity lexicon, emotional category features, emotional dimension features or structural features.

In recent years, deep learning-based approaches have been applied to irony detection, where (deep) features are automatically derived from texts using neural network models. Using the similarity score between word embeddings as features has shown improvement for irony detection (Joshi et al., 2016). In a previous study (Nozza, Fersini, & Messina, 2016), an unsupervised framework for domain independent irony detection, which takes the advantages of probabilistic topic models for discovering topic-irony and meanwhile using word embeddings to improve the generalization abilities. A convolutional neural network (CNN) was proposed in Poria et al. (2016) for irony detection, which uses a pre-trained convolutional neural network for extracting sentiment, emotion and personality features for irony detection. There are also several studies that use CNN-LSTM structures (Ghosh & Veale, 2016; 2017) for sarcasm detection. Another interesting work focuses on detecting rhetorical questions and sarcasm using CNN-LSTM also, but with an additional fully connected layer used for the purpose of taking Linguistic Inquiry and Word Count (LIWC) features (Oraby et al., 2017). These existing studies use the convolutional network to automatically derive deep features from texts for irony detection. Results of these deep learning approaches are generally better than classical feature engineering-based approaches.

Recently attention-based recurrent neural networks (RNNs) were proposed for irony detection (Felbo, Mislove, Søgaard, Rahwan, & Lehmann, 2017; Ghosh et al., 2017; Huang et al., 2017) and other NLP tasks (Luong, Pham, & Manning, 2015; Wang, Huang, & Zhao, 2016; Zhou et al., 2016). The self-attention mechanism is not directly targeted to identify context incongruity. Some of the previous work (Felbo et al., 2017) studied emotion, sentiment and sarcasm prediction, where the attention mechanism is not particularly used to detect context incongruity. Some other previous work (Ghosh et al., 2017) studied irony detection for replies in social media conversions. The sentence-level attention mechanism is used to identify more informative sentences in conversations that trigger sarcasm replies. In addition, a previous study (Huang et al., 2017) focused on irony detection in tweets and employed the standard attention mechanism. However, the standard self-attention mechanism often generates attentions for only partial texts forming the context-statement contrast and thus fail to detect the context incongruity (More details in Section 3). A recent study proposed a neural network with intra-attention for sarcasm detection on social media, which is focusing on intricate similarities between each word pair in sentence (Tay et al., 2018). Almost all of the previous studies are using a handful human-labeled ironic tweets for training, however, pattern recognition for detecting irony is so complex and difficult that a considerable size of the dataset is needed. As it is costly to build a large annotated dataset for training a high-performance model, transfer learning with sufficient sentiment resources seems to be at hand as an alternative.

Irony detection via identifying context incongruity has been reported in the literature, but the proposed solutions very much rely on manually engineered patterns and features. In one of the previous work (Riloff et al., 2013), sarcasm is identified via a pattern of "positive sentiment followed by negative situation", and a bootstrapping algorithm (originated from the word "love") to automatically learn phrases corresponding to the positive sentiment and negative situation respectively. In Joshi et al. (2015), four types of manually engineered features including lexical features, pragmatic features, implicit incongruity features and explicit incongruity features are used to train a model for irony detection. It must be noted that although irony detection needs to detect sentiment incongruity, it is different from detecting sentiment shift (Xia, Xu, Yu, Qi, & Cambria, 2016), where words and phrases change the sentiment orientation of texts as in "I don't like this movie".

Existing studies (Poria et al., 2016) that use sentiment analysis resources for irony detection lacks a principled approach of transferring the sentiment analysis knowledge. In Poria et al. (2016), a comprehensive set of features, including sentiment, emotion and personality features, are extracted from sentiment analysis resources for irony detection. The model combines all features before the prediction layer in the neural network, which makes it unclear whether the sentiment features benefit detecting context incongruity or irony detection. In contrast, our work does not only use sentiment knowledge but also consider the reasoning of how and why we incorporate them in a neural network. Specifically, we devoted to using sentiment knowledge and resources with reasoning and visualization-focused interpretation to show how our models detecting context incongruity.

4. Attention-based Bi-LSTM (Bi-LSTM)

In this section, we introduce attention-based Bi-LSTM for the sake of understanding our proposed models. Recurrent neural networks (RNNs) are designed to process sequences. The Long Short-Term Memory (**LSTM**) is a commonly used RNN unit proposed by Hochreiter and Schmidhuber (1997) to overcome the gradient vanishing problem. In terms of the network architecture, the Bidirectional LSTM (Zhou et al., 2016) is widely used, which has two layers of LSTM reading sequences forward and backward respectively. The output of Bi-LSTM is a concatenation of forward and backward returned sequences:

$$h_i = [\overrightarrow{h_i} \| \overleftarrow{h_i}] \tag{1}$$

In the attention-based Bi-LSTM, $H = [h_1, h_2, \ldots, h_i]$ is a matrix consisting of output vectors produced by the Bi-LSTM, where i is the time step. The representation r of a tweet is formed by a weighted sum of these output vectors:

$$M = \tanh(H) \tag{2}$$

$$\alpha = softmax(\omega^T M) \tag{3}$$

$$r = H\alpha^T \tag{4}$$

$$h^* = \tanh(r) \tag{5}$$

At prediction, we use *softmax* to predict $\hat{y}$ for a tweet. The goal of training is to minimize the cross-entropy error between the true label y_i and the predicted label $\hat{y}_i$:

$$\hat{y} = softmax(Wh^* + b) \tag{6}$$

$$loss^1 = -\sum_i \sum_j y_i^j \log \hat{y}_i^j \tag{7}$$

In Eq. (3), the vector α is the attention vector.

Bi-LSTM often fails to capture words and phrases crucial for building the ironic intent. This may be due to the inherent difficulty of the task and limited annotated training instances.

As shown in Fig. 2, with the first tweet “someone needs stop me before kill someone ☺ love waking up in the worst fcking mood”, Bi-LSTM only put strong attention on “loving waking” that indicates positive sentiment, and as a result failed to detect the negative sentiment expressed by “the worst fcking mood”.

The failure of the standard attention mechanism for detecting context incongruity is possibly caused by the inherent difficulty of the task and only relying on the irony labels, which are limited. Moreover, LSTM even with attention can only learn the long dependencies of the context. To detect context incongruity, we need to use external sentiment resources. In the next section, we describe our approach of enhancing the attention mechanism with sentiment knowledge transferred from the readily available resources for sentiment analysis.

5. Sentiment-based transfer learning for irony detection

Transfer learning is an important machine learning technique that takes advantages of knowledge from solving one problem to solve other related problems, which can overcome the burden of limited human-labeled resources. Learning deep features or abstract representation of input is the advantage of deep learning used with transfer learning (Bengio, 2012). Transfer learning based models are particularly useful for cross-domain tasks. Especially when a target domain has very limited data, there is a need to train a high-performance model using data in a source domain where data can be easily obtained (Weiss, Khoshgoftaar, & Wang, 2016). Feature transformation can be completed by re-weighting a layer in the source domain to more closely match the target domain (Cao, Li, Li, & Wei, 2017), or by mapping features from both source domain and target domain into a common latent feature space (Shu, Qi, Tang, & Wang, 2015).

In our scenario, since detecting irony on Twitter is based on incongruous sentiment between the statement and the context, knowledge learned from the resources used for sentiment analysis will be incorporated into detecting irony. Sentiment analysis resources are widely available, including sentiment word corpora (Baccianella, Esuli, & Sebastiani, 2010; Hutto & Gilbert, 2014) and sentiment tweets corpora (Ghosh et al., 2015; Rosenthal, Farra, & Nakov, 2017). To improve the attention mechanism on detecting context incongruity, we propose our methods that are transferring sentiment knowledge from external resources, such as sentiment words corpora and sentiment Twitter corpus (Baccianella et al., 2010; Hutto & Gilbert, 2014; Wilson et al., 2005), as additional resources to enrich the limited human annotated ironic tweets. The challenge is how to represent and incorporate the sentiment knowledge into the attention mechanism for irony detection.

In order to incorporate two different types of sentiment resources into irony detection, namely sentiment word lexica and

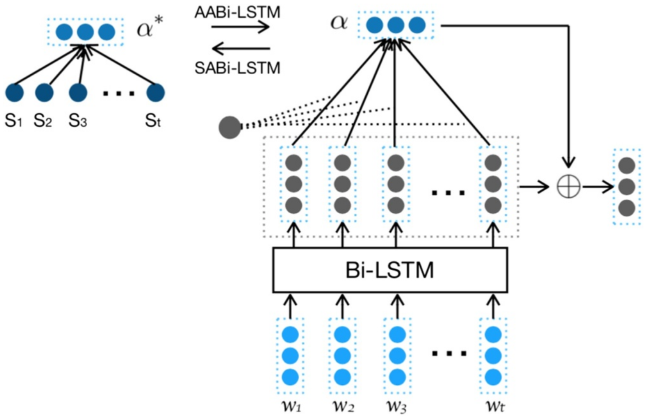


Fig. 1. Sentiment Attention Bi-LSTM models. AABi-LSTM: model combines the hard sentiment attention with the learned soft attention. SABi-LSTM: model treats the hard sentiment attention as a supervised signal.

sentiment tweets copra, we propose different models to transfer different sentiment knowledge. The first two models are incorporating sentiment word lexica, where the sentiment-based hard attention is generated to strengthen the attention distribution on sentiment parts, but with different methods. The major difference between them is how the sentiment-based hard attention being incorporated. In the first model, the sentiment-based hard attention is treated as a feature while it is treated as a supervised signal in the second model. Being different from our first two models, our third model is proposed to detect context incongruity using the transferred deep features from the model learned on sentiment Twitter corpus instead of using the sentiment-hard attention.

5.1. Sentiment-augmented attention Bi-LSTM (AABi-LSTM)

With the first model, the readily available sentiment word corpora (Baccianella et al., 2010; Hutto & Gilbert, 2014; Wilson et al., 2005) is used as additional resources to generate a sentiment distribution, which then will be treated as an hard attention and transferred into the soft attention mechanism in order to push the attention-based model to focus on context incongruity. In particular, our model incorporates not only the polarity but sentiment strength into the attention mechanism to capture the strength of incongruity in tweets, based on the linguistic principle of “the extent of irony perception depends on the strength of context and statement” (Gerrig & Goldvarg, 2000; Ivanko & Pexman, 2003).

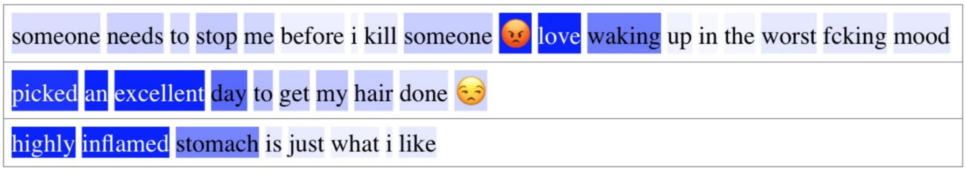
In our model AABi-LSTM as shown in Fig. 1, we first construct a sentiment hard attention based on the sentiment of each word. The sentiment scores of each word are generated by using pre-defined sentiment corpora. For a given tweet, the sentiment distribution $[\alpha_1^*, \alpha_2^*, \alpha_3^*, ..., \alpha_i^*]$ is generated by applying *softmax* on absolute sentiment scores of each word $[S_1, S_2, S_3, ..., S_i]$. Then, we transfer this sentiment hard attention into the attention-based model to enhance the attention to the sentiment part of a tweet. In our first proposed mechanism, the sentiment attention vector is weighted and then added to the learned attention vector in the network, which results in directly strengthening the attention of the network on the sentiment part:

$$\alpha^* = softmax(|S|) \tag{8}$$

$$r = H(\alpha \oplus W\alpha^*)^T \tag{9}$$

5.2. Sentiment-supervised attention Bi-LSTM (SABi-LSTM)

In order to detect the complete contextual incongruity, we further propose to take advantage of the widely available sentiment Twitter corpora (Ghosh et al., 2015; Rosenthal et al., 2017) to improve the attention mechanism in a supervised manner so as to capture the complete context for incongruity. With our second model sentiment-supervised attention Bi-LSTM (SABi-LSTM), we learn the abstract representation of polarity embedded in expressions without sentiment words and transfer these learned features into irony detection model for learning context incongruity.


someone needs to stop me before i kill someone 😡 love waking up in the worst fcking mood

picked an excellent day to get my hair done 😒

highly inflamed stomach is just what i like

Fig. 2. Examples of attention generated by the standard attention-based Bi-LSTM; the luminance of blue represents the attention value of each word. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

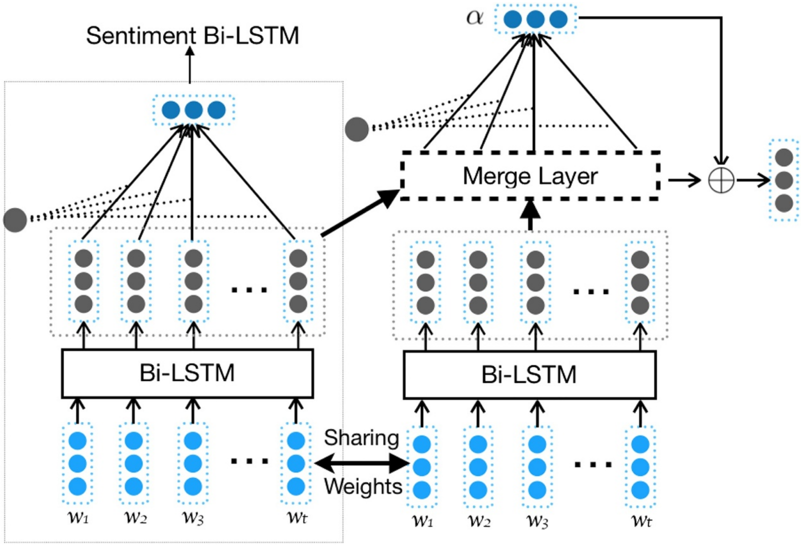


Fig. 3. Sentiment transferred model (STBi-LSTM): has two training steps. 1. The sentiment Bi-LSTM is firstly trained on a sentiment corpus. 2. Two Bi-LSTMs are trained together on an irony corpus, but with the weight of sentiment Bi-LSTM frozen.

As shown in Fig. 1, SABi-LSTM includes a sentiment attention mechanism that uses the sentiment hard attention in a supervised manner, which provides an alternative supervised signal to let the model learn features and attentions with reinforced attentions on sentiment parts. Technically, the attention value is used as an output of the model apart from class prediction, which will be then used in supervised training with sentiment hard attention as the true label. In order to let the network's attention be close to sentiment distribution, another loss function is defined to minimize the *cosine distance* or (1 − *Cosine_Similarity*) between attention distribution and sentiment distribution, as follows:

$$loss^2 = 1 - \frac{\sum_{i=1}^{T} \alpha_i \alpha_i^*}{\sqrt{\sum_{i=1}^{T} \alpha_i^2}\sqrt{\sum_{i=1}^{T} (\alpha^*)_i^2}} \tag{10}$$

$$loss = loss^1 + \lambda^* loss^2 \tag{11}$$

λ is a hyper-parameter to adjust $loss^2$ when updating neural network.

5.3. Sentiment transferred Bi-LSTM (STBi-LSTM)

The previous two proposed models are transferring sentiment hard attention. Our third proposed method illustrated in Fig. 3 is designed to transfer deep features from sentiment analysis into irony detection for learning both explicit and implicit context incongruity. Our model consists of two Bi-LSTMs. one of Bi-LSTM acts as the sentiment feature extractor, while another one is the irony detector. The training process contains two parts. Firstly, the sentiment Bi-LSTM is trained on a readily available Twitter sentiment corpus, and then the weights of Bi-LSTM are kept frozen. In the second part of the training, a tweet will be given to both Bi-LSTMs. The sentiment Bi-LSTM (or the sentiment feature extractor) outputs deep features that are about words with the implicit and explicit sentiment, and the second model firstly learns semantic features for the context. Both features are then mapped into a common latent feature space at Merger layer, and features on incongruous context are learned by the attention layer and the fully connected layer of the second Bi-LSTM. In terms of the mathematical operation of incorporating transferred deep features at Merger layer, it concatenates deep features from sentiment Bi-LSTM and the second Bi-LSTM before the attention mechanism:

$$H_{merged} = [H_{semantic} || H_{sentiment}] \tag{12}$$

6. Experiments

We next discuss the experiment setup, including baselines and datasets, and then report results. We also report on results of error analysis for our models.

6.1. Baselines and datasets

We compared our models against deep learning-based irony detection models as well as representative conventional feature-based models.

- Bi-LSTM: Attention-based Bi-LSTM structure has been employed for irony detection in conversations and for learning representations for irony detection in the literature (Felbo et al., 2017; Ghosh et al., 2017). We implemented the network for our task and our implementation is based on the popular structure in Zhou et al. (2016).
- CNN-LSTM: Our implementation closely followed the architecture in Ghosh and Veale (2016). It has three different neural network layers, a convolutional layer followed by 2 LSTM layers and a fully connected layer with the same hyper-parameter settings.
- LSTM: The model proposed in Huang et al. (2017) is an attention-based LSTM for irony detection on Twitter.
- CNN: The Convolutional network (Kim, 2014) is widely used for classification problems.
- Riloff et al. (2013), Joshi et al. (2015) and Hernańdez et al. (2016) are classical feature-based irony detection models. Especially (Riloff et al., 2013) and (Joshi et al., 2015) are representative models focused on context incongruity.

Several datasets are widely used in the irony detection literature. There are two approaches to annotate sarcasm. Some datasets are automatically annotated by using sarcasm hashtags #irony, #sarcasm and #not. Other datasets are manually annotated by humans.

- Reyes2013 (Reyes, Rosso, & Veale, 2013), Barbieri2014 (Barbieri, Saggion, & Ronzano, 2014) and Ptacek2014 (Ptáček, Habernal, & Hong, 2014) are datasets automatically annotated by hashtags as shown in Table 1.
  Each pair of sarcasm and non-sarcasm class of tweets form a dataset for evaluating irony detection.
- Riloff2013 (Riloff et al., 2013), Moh2015 (Mohammad, Zhu, Kiritchenko, & Martin, 2015) and SemEval2018 (Hee, Lefever, & Hoste, 2018) are manually annotated Twitter datasets. SemEval2018 is the official dataset used for SemEval 2018 Task 3 (Irony detection in English tweets). Statistics of the datasets are shown in Table 1.
- For an interesting comparison between hashtag-labeled datasets and manually labeled datasets, we used hashtags, such as #sarcasm and #irony, to automatically label SemEval2018, which ended up with one more manually labeled dataset so that we can compare our models on a dataset with different annotation strategies. The details of the dataset are shown in Table 1.

For each dataset, we randomly split it into 80% for training and 20% for testing, except SemEval2018 using official training and testing splits. The parameters are tuned on 10% random portion of the training data. For a fair comparison, following the literature, macro average F_1 was used as the evaluation metric, except for SemEval2018 where the binary F_1 was adopted.

Word-level tokenization is using the tokenizer from spaCy.[1] The word embeddings for all models have been initialized with the pre-trained Fasttext (Mikolov, Grave, Bojanowski, Puhrsch, & Joulin, 2018) word vectors with 300 dimensions. The word-level sentiment scores are generated by NLTK with the help of a sentiment analysis tool VADER (Hutto & Gilbert, 2014), which is designed for sentiment analysis of social media data, especially Twitter. Another great advantage of VADER is that it not only provides the polarity of words, but also gives the sentiment strength of words. Also, we adopted a sentiment emoji corpus (Novak et al., 2015). The sentiment corpus for transfer learning used in our STBi-LSTM, is built based on two sentiment corpora used in SemEval 2017 Task 4 (Rosenthal et al., 2017) and SemEval2015 Task 11 (Ghosh et al., 2015). The hyper-parameters are selected using a grid search. The best dimensions of hidden states for all variants of Bi-LSTMs in our grid search is 200.

6.2. Evaluation

In order to clearly show how models perform on datasets using different annotation strategies, we report experiment results in two separate tables. Table 2 reports the experimental results on datasets labeled using hashtags #irony or #sarcasm. Table 3 reports the experiment results on datasets annotated by humans on crowdsourcing platforms. Results for the hashtag-labeled SemEval2018 dataset are also included in Table 3 for easy comparison with results for the manually labeled SemEval2018. Additionally, we reported the results of our models on class-unbalanced datasets in Table 4.

According to Table 2, it is obvious that irony detection on hashtag-annotated datasets is not as difficult as on manually labeled dataset. Most models have achieved very promising results almost across all of hashtag-labeled datasets. Moreover, our results are also consistent with a previous study (Joshi et al., 2015), where experiments show that models in their study have better performance on hashtag-labeled datasets than that of manually annotated datasets. Additionally, neural models perform better than traditional machine learning models, such as feature engineering with SVM (Hernańdez et al., 2016).

Table 3presents experiment results on human annotated datasets. In general, attention-based models work better than other models including other neural models and conventional machine learning methods. On dataset Riloff2013, the proposed three models all achieved better results than the baselines, especially our third proposed model achieved the best result which has about 4% improvement over the Bi-LSTM.

In our experiments, the dataset Moh2015 is the most difficult one, which is not only because of the relatively small size of the dataset (1397 non-ironic tweets and 532 ironic tweets), but also because of the type of irony expressed in this dataset. This dataset was collected using hashtags related to the “2012 US presidential election”, so that most of the ironic tweets are either situational irony or ironic utterance related to named entity or external knowledge.

For example, in Fig. 4, in order to detect irony expressed by those tweets, a model has to have the knowledge of named entity, such as “Obama” and “Romney”, or background knowledge of the tasked event “2012 US presidential election”. However, our model

[1] https://spacy.io.

Table 1
Datasets.

	Ironic vs. Non-ironic	Annotation
Reyes13	10,000 vs. 10,000	#irony vs. #education, #humor, or #politics
Barbieri14	10,000 vs. 10,000	#irony, #sarcasm vs. #eduation, #humor, etc.
Ptacek2014	18,889 vs. 48,890	#sarcasm for sarcastic tweets
SemEval2018_Hashtag	2826 vs. 1792	#sarcasm, #irony, #not
Riloff2013	474 vs. 1608	manual
Moh2015	532 vs. 1397	manual
SemEval2018	2222 vs. 2396	manual

Table 2
Results (F_1) for Irony detection on hastag-annotated datasets.

	Reyes2013			Barbieri2014			
	edu	*hum*	*pol*	*edu*	*hum*	*pol*	*news*
Bi-LSTM	94.01	95.54	96.32	94.12	94.86	98.62	96.24
CNN-LSTM	92.04	92.73	93.33	93.15	94.78	97.48	96.10
LSTM	92.30	89.50	89.00	94.03	94.23	97.56	96.11
CNN	93.35	93.44	94.66	94.12	95.53	98.31	96.28
AABi-LSTM	94.23	95.56	96.42	94.92	95.73	98.18	96.91
SABi-LSTM	94.65	**95.82**	96.15	94.21	95.16	98.34	96.41
STBi-LSTM	**94.69**	95.69	**96.55**	**94.95**	**96.14**	**98.62**	**96.92**
Hernańdez et al. (2016)[a]	90.00	90.00	92.00	90.00	92.00	94.00	96.00

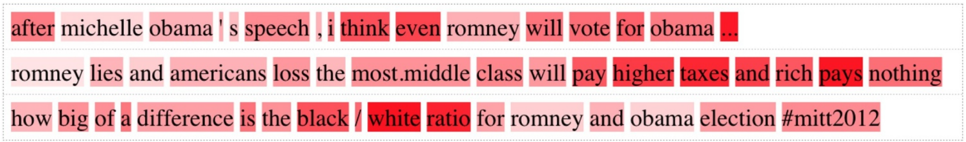
[a] FAs reported in the relevant papers.

Table 3
Results (F_1) for Irony detection on manually annotated datasets.

	Riloff2013	Moh2015	SemEval2018	SemEval2018 Hashtag-labeled
Bi-LSTM	73.57	58.31	64.15	69.08 / 72.18
CNN-LSTM	70.56	59.22	61.16	67.21 / 70.52
LSTM	72.17	57.16	63.66	67.04 / 72.47
CNN	74.75	57.71	62.03	70.49 / 72.80
AABi-LSTM	75.39	61.54	67.86	71.85 / 73.28
SABi-LSTM	74.63	63.37	65.33	70.25 / 73.05
STBi-LSTM	**77.85**	**66.80**	69.00	**72.11** / **73.56**
Reported Best Result	73.00 Hernańdez et al. (2016)	66.00 Hernańdez et al. (2016)	**70.54** Hee et al. (2018)	-

Table 4
Results (F_1) for Irony detection on class-unbalanced datasets.

	Reyes_Unbalanced (10,000 vs. 30,000)	Barbieri_Unbalanced (10,000 vs 40,000)	Ptacek2014 (18,889 vs. 48,890)
Bi-LSTM	92.15	95.22	98.12
CNN-LSTM	91.64	95.06	98.00
LSTM	91.23	94.59	98.21
CNN	92.06	95.14	98.62
AABi-LSTM	91.70	95.30	98.83
SABi-LSTM	92.15	95.21	98.52
STBi-LSTM	**92.25**	**95.66**	**99.05**

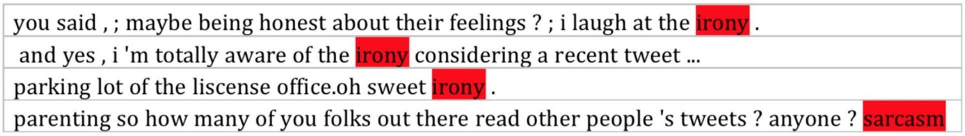


after michelle obama ' s speech , i think even romney will vote for obama ...

romney lies and americans loss the most.middle class will pay higher taxes and rich pays nothing

how big of a difference is the black / white ratio for romney and obama election #mitt2012

Fig. 4. Examples of attention distribution on ironic tweets from Moh2015. Tweets in red are incorrectly classified. Attention are generated by our model STBi-LSTM. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



you said , ; maybe being honest about their feelings ? ; i laugh at the irony .

and yes , i 'm totally aware of the irony considering a recent tweet ...

parking lot of the liscense office.oh sweet irony .

parenting so how many of you folks out there read other people 's tweets ? anyone ? sarcasm

Fig. 5. Examples of hashtag-labeled ironic tweets in Reyes2013.

(STBi-LSTM) is still able to capture most of the words or phrases with sentiment meaning. The dataset used by SemEval 2018 Task 3 "irony detection in English tweets" is another difficult task. Our proposed models have achieved better results than other baseline neural models, and our results are also comparable to the best in the official rank (Hee et al., 2018).

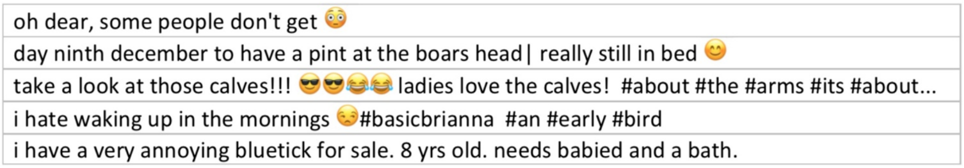
In terms of statistical significance, Wilcoxon signed rank tests (Wilcoxon, 1945) indicated that the accuracy was significantly higher in our model STBi-LSTM (Mdn = 95.31) than in LSTM (Mdn = 91.77), $Z = 3.67$, $p < .001$, $r = .98$; CNN (Mdn = 93.78), $Z = 2.83$, $p < .002$, $r = .76$; CNN-LSTM (Mdn = 92.94), $Z = 3.67$, $p < .001$, $r = .98$; BiLSTM (Mdn = 94.49), $Z = 2.94$, $p < .002$, $r = .78$.

As we discussed before, results of all of our models achieved better performance on hashtag-labeled datasets than that of manually labeled datasets. There are several reasons. First of all, the size of the dataset could affect the performance of neural models. The hashtag-labeled datasets have a greater number of tweets than that of manually labeled datasets, for example Reyes2013 (Irony vs Education) has about twenty thousand tweets which is ten times of the number of tweets in Moh2015. Deep learning models give better results on the hashtag-based datasets in comparison to the manually-tagged ones because they are bigger, and deep learning models succeed in generalizing if data is big enough. Furthermore, the labeling approach used for manually annotated datasets is also one of the factors that influences the performance of deep learning models. For example, in Reyes2013 datasets, the ironic datasets were annotated by the appearance of #irony while the non-ironic datasets were labeled by the appearance of #education, #humor or #politics. In contrast, non-ironic datasets in manually annotated datasets are most likely to be general tweets (SemEval2018, Riloff2013) or a particular event related (Moh2015).

Lastly but most importantly, we found that all of attention-based models tend to put high attention on a list of common words. For example in Fig. 5, words like "irony" or "sarcasm" in Reyes2013, are highly attended by attention-based models. It must be noted that ironic tweets were automatically labeled by the hashtag "#irony", which has been removed from ironic tweets. In these tweets, the word "irony" is acting like a topic word or an important part of the context. Removing the word "irony" is more likely to make a context nonsense or incomplete. We have done further analysis on ironic tweets in Reyes2013 and found that the frequencies of the word "irony" is about 3531 in ten thousand ironic tweets. Moreover, even for non-ironic tweets, there are also some common topic words, for example "education", "technology", "science", "edchat" etc. in Reyes2013 (Irony vs Education). All of the deep learning models achieved very good results on hashtag-labeled datasets by learning patterns based on common topic words, such as "irony" and "technology", appeared in the context. However, all attention-based models failed to detect the context incongruity since all of them put attentions mainly on common topic words. The performance improvement of our models over Bi-LSTM on hashtag-labeled datasets is not as pronounced as for the manually-labeled datasets may be explained by that attention-based models are strongly affected by these common topic words in the datasets instead of the context incongruity.

In Table 3, for the last column "SemEval2018 Hashtag-labeled", there are two results for each model. Results on the left side are produced from experiments using the hashtag-labeled training data but the manually annotated testing data, while results on the right side are produced by experiments using both the hashtag-labeled training and testing dataset. Comparing the columns SemEval2018" and "SemEval2018 Hashtag-labeled", all models have better performance using the hashtag-labeled training datasets than that using manually labeled datasets. Most interestingly, using hashtag-labeled SemEval2018 training data improves performance of all neural models on manually labeled SemEval2018 testing data, which demonstrates that the hashtag labels contain supervision signals for irony detection. On the other hand, models have the best performance for hashtag-labeled training and test data, which demonstrates that hashtags define some notion for sarcasm that can be more easily identified. Indeed we analysed some randomly selected tweets labeled as ironic by hashtags but are labeled as non-ironic by humans and found that most of them have obvious sentiment patterns. Some example tweets are shown in Fig. 6.

Table 4shows the performance of our models on class-unbalanced datasets. Generally, the proposed model STBi-LSTM works the best among all of models. However, the performance difference among models are very tiny. In terms of performance difference between the class-balanced dataset and the class-unbalanced dataset, the results of all models are worse on unbalanced datasets than that of on balanced datasets by around 2.21 (Reyes2013) and 0.83 (Barbieri2014).



oh dear, some people don't get 😳

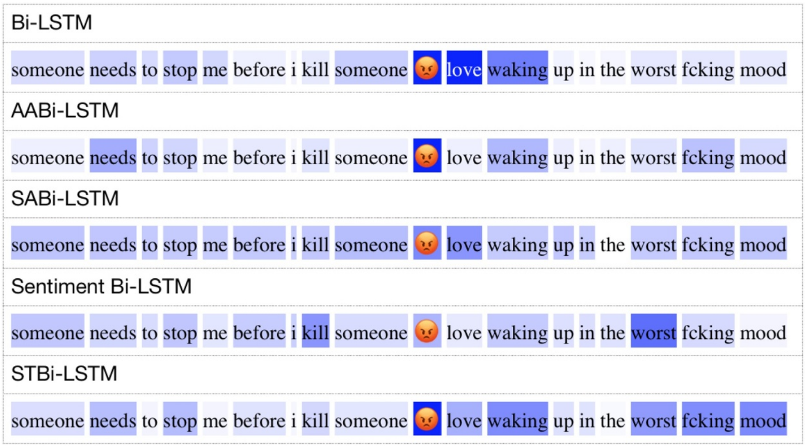
day ninth december to have a pint at the boars head| really still in bed 😊

take a look at those calves!!! 😎😎😂😂 ladies love the calves! #about #the #arms #its #about...

i hate waking up in the mornings 😒#basicbrianna #an #early #bird

i have a very annoying bluetick for sale. 8 yrs old. needs babied and a bath.

Fig. 6. Examples of hashtag-labeled ironic tweets, while these tweets were annotated as non-ironic by human. Labeling hashtags #sarcasm, #irony, or #not are not included.



Bi-LSTM

someone needs to stop me before i kill someone 😡 love waking up in the worst fcking mood

AABi-LSTM

someone needs to stop me before i kill someone 😡 love waking up in the worst fcking mood

SABi-LSTM

someone needs to stop me before i kill someone 😡 love waking up in the worst fcking mood

Sentiment Bi-LSTM

someone needs to stop me before i kill someone 😡 love waking up in the worst fcking mood

STBi-LSTM

someone needs to stop me before i kill someone 😡 love waking up in the worst fcking mood

Fig. 7. Differences of attention distribution among attention-based models.

6.3. Discussion

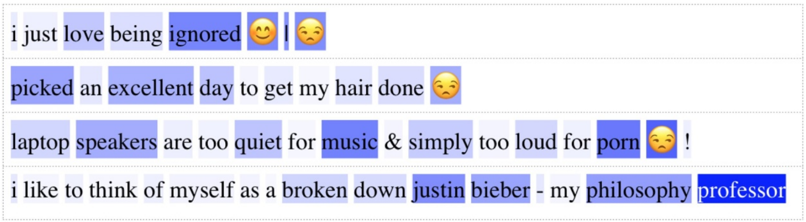
We first discuss how our models improve the attention mechanism for detecting contexts for sentiment contrast. We further discuss how our third proposed model (STBi-LSTM) learns the contexts for explicit and implicit incongruity.

Fig. 7presents an ironic tweet that has differently learned attention by our models and Bi-LSTM. Generally, Bi-LSTM is able to detect some of the words with sentiment meaning, but it seems often fail to detect the full context for incongruity. For example, in the first example of Fig. 7, it spreads very high attention on phrase “love waking up” expressing the positive sentiment, but with very tiny attention on phrase “worst fcking mood” which is the negative situation of this ironic tweet. In contrast, our proposed models all have more attention on “worst fcking mood”. Most importantly, without using an explicit lexicon, our third proposed model STBi-LSTM is still able to detect both parts of context incongruity and spread balanced attention on both.

With the transferred deep features from the sentiment model, the STBi-LSTM performs very well, especially on detecting sentiment based context incongruity. We selected several examples of attention learned by STBi-LSTM in Fig. 8. In this figure, the first two tweets are examples of explicit context incongruity. “love” versus “ignored” is the key sentiment contrast in the first example, while “excellent day” versus “☹” is the sentiment contrast in the second example. In order to show the ability of our models on detecting implicit context incongruity, we picked two example tweets from the dataset. In Fig. 8, the third tweet is ironic about the sound of laptop speaker, and the irony is expressed by contrasting two situations, which are “quiet for music” and “loud for porn”. Each of these two phrases does not have the explicit sentiment until they are in a contrasting context, and our model has successfully identified most key patterns for building the implicit context incongruity. The last example has two named entities as the context incongruity, which does not really have sentiment meaning until our model pass the learned sentiment knowledge to them. Both “Justin Bieber” and “philosophy professor” appear a few times in our sentiment training corpus, and tweets with “Justin Bieber” are more likely to be negative while tweets with “philosophy professor” are more likely to be positive. Even though both named entities do not have sentiment meaning in general, the supervised sentiment training can embed an implicit sentiment via deep features. With the implicit sentiment features learned in sentiment training, our irony model STBi-LSTM successfully detects the context incongruity at the second stage of learning.

6.4. Error analysis

Our models can only detect irony based on the self-contained contents in tweets. In order to understand what our models miss on irony detection, we provide several mistakenly classified examples with their original version of text content in Fig. 9. In ironic tweets, hashtags such as #irony, #sarcasm and #not are often used to indicate the irony intention. When these hashtags are removed for learning a more general model, it is hard to imagine that even humans can classify such tweets as ironic. For example, the second



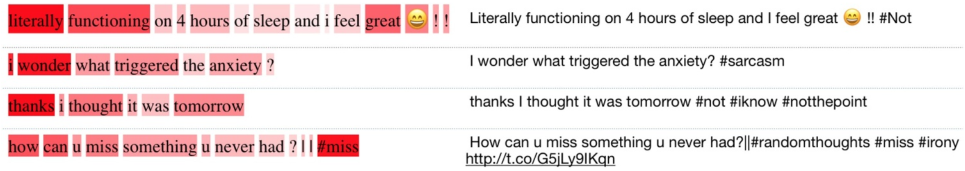
i just love being ignored 😊 | 😒

picked an excellent day to get my hair done 😒

laptop speakers are too quiet for music & simply too loud for porn 😒 !

i like to think of myself as a broken down justin bieber - my philosophy professor

Fig. 8. Examples of attention distribution learned by STBi-LSTM.



literally functioning on 4 hours of sleep and i feel great 😄 ! !	Literally functioning on 4 hours of sleep and I feel great 😄 !! #Not
i wonder what triggered the anxiety ?	I wonder what triggered the anxiety? #sarcasm
thanks i thought it was tomorrow	thanks I thought it was tomorrow #not #iknow #notthepoint
how can u miss something u never had ? | | #miss	How can u miss something u never had?||#randomthoughts #miss #irony http://t.co/G5jLy9IKqn

Fig. 9. Examples of mistakenly classified tweets (ironic tweets have been classified as non-ironic): tweets in the left column are chosen from the SemEval2018 test data, tweets in the right column are their original version where all hashtags are kept. (The luminance of red represents the attention value of each word paid by our STBi-LSTM model). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

and third examples, none of them carry the irony sense when the hashtags “#sarcasm” and “not” is removed.

Some irony can only be inferred from the conversational context. As a result, when the complete conversational context is not available, it is even hard for a human to find the irony utterance (Ghosh et al., 2017; Riloff et al., 2013). In our examples, the second and third tweets are more likely coming from a conversational context where the authors of these tweets wrote them to express ironic intent. In the first example, our model successfully detected the positive sentiment “feel great”, but failed to detect the negative situation “4 hours of sleep”.

7. Conclusion

In this paper, we studied the problem of irony detection on Twitter. Context incongruity is a commonly seen form of irony on Twitter, where the contrast between the positive statement and the negative context is the common form of context incongruity. We proposed to employ transfer learning and attention-based neural network to identify context incongruity for detecting irony. The most challenging part for training a good automatic irony detection model is the limited human labeled dataset. In contrast with irony detection, sentiment analysis has sufficient resources, such as pre-defined sentiment lexica and human annotated corpora. We proposed our models to take advantage of these widely and readily available sentiment resources to improve the ability of attention-based model on detecting context incongruity. With incorporating transferred sentiment, our models are able to detect both implicit and explicit context incongruity at most times. Experiments show that our three proposed sentiment attention mechanisms result in better performance than the baselines including several popular neural models for irony detection on Twitter.

Acknowledgments

The authors would like to express their gratitude to the anonymous reviewers for their helpful comments.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ipm.2019.04.006.

References

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). *Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. LRECvol. 10. LREC* 2200–2204.

Barbieri, F., Basile, V., Croce, D., Nissim, M., Novielli, N., & Patti, V. (2016). *Overview of the evalita 2016 sentiment polarity classification task. Proceedings of third Italian conference on computational linguistics (CLiC-it 2016) & fifth evaluation campaign of natural language processing and speech tools for Italian. Final workshop (EVALITA 2016)*.

Barbieri, F., Saggion, H., & Ronzano, F. (2014). *Modelling sarcasm in Twitter, a novel approach. Proceedings of the 5th workshop on computational approaches to subjectivity, sentiment and social media analysis*50–58.

Bengio, Y. (2012). *Deep learning of representations for unsupervised and transfer learning. Proceedings of ICML workshop on unsupervised and transfer learning*17–36.

Cao, Z., Li, W., Li, S., & Wei, F. (2017). *Improving multi-document summarization via text classification. Proceedings of AAAI*3053–3059.

Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., & Lehmann, S. (2017). *Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. Proceedings of the 2017 conference on empirical methods in natural language processing*1615–1625.

Gerrig, R. J., & Goldvarg, Y. (2000). Additive effects in the perception of sarcasm: situational disparity and echoic mention. *Metaphor and Symbol, 15*(4), 197–208.

Ghosh, A., Li, G., Veale, T., Rosso, P., Shutova, E., Barnden, J., et al. (2015). *Semeval-2015 task 11: sentiment analysis of figurative language in Twitter. Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*470–478.

Ghosh, A., & Veale, T. (2016). *Fracking sarcasm using neural network. Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*161–169.

Ghosh, A., & Veale, T. (2017). *Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal. Proceedings of the 2017 conference on empirical methods in natural language processing*482–491.

Ghosh, D., Fabbri, A. R., & Muresan, S. (2017). *The role of conversation context for sarcasm detection in online interactions. Proceedings of the 18th annual SIGdial meeting on discourse and dialogue*186–196.

González-Ibáñez, R., Muresan, S., & Wacholder, N. (2011). *Identifying sarcasm in Twitter: A closer look. Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*581–586.

Hee, C. V., Lefever, E., & Hoste, V. (2018). *Semeval-2018 task 3: Irony detection in English tweets. Proceedings of the 12th international workshop on semantic evaluation*39–50.

Hernańdez, F. D. I., Patti, V., & Rosso, P. (2016). Irony detection in Twitter: The role of affective content. *ACM Transactions on Internet Technology, 16*(3), 19.

Hernańdez, F. D. I., & Rosso, P. (2016). Irony, sarcasm, and sentiment analysis. In F. A. Pozzi, E. Fersini, E. Messina, & B. Liu (Eds.). [*Sentiment analysis in social networks*]. Morgan Kaufmann Ch. 7

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation, 9*(8), 1735–1780.

Huang, Y. H., Huang, H. H., & Chen, H. H. (2017). *Irony detection with attentive recurrent neural networks. Proceedings of European conference on information retrieval.* Springer534–540.

Hutto, C., & Gilbert, E. (2014). *Vader: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of eighth international AAAI conference on weblogs and social media*.

Ivanko, S. L., & Pexman, P. M. (2003). Context incongruity and irony processing. *Discourse Processes, 35*(3), 241–279.

Joshi, A., Bhattacharyya, P., & Carman, M. J. (2017). Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR), 50*(5), 73.

Joshi, A., Sharma, V., & Bhattacharyya, P. (2015). *Harnessing context incongruity for sarcasm detection. Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing, vol.* 2757–762.

Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P., & Carman, M. (2016). *Are word embedding-based features useful for sarcasm detection? Proceedings of the 2016 conference on empirical methods in natural language processing*1006–1011.

Kim, Y. (2014). *Convolutional neural networks for sentence classification. Proceedings of the 2014 conference on empirical methods in natural language processing*1746–1751.

Kunneman, F., Liebrecht, C., Mulken, M. V., & Bosch, A. V.d. (2015). Signaling sarcasm: From hyperbole to hashtag. *Information Processing & Management, 51*(4), 500–509.

Luong, T., Pham, H., & Manning, C. D. (2015). *Effective approaches to attention-based neural machine translation. Proceedings of the 2015 conference on empirical methods in natural language processing*1412–1421.

Maynard, D., & Greenwood, M. A. (2014). *Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. LREC*4238–4243.

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., & Joulin, A. (2018). *Advances in pre-training distributed word representations. Proceedings of the international conference on language resources and evaluation (LREC 2018)*.

Mohammad, S. M., Zhu, X., Kiritchenko, S., & Martin, J. (2015). Sentiment, emotion, purpose, and style in electoral tweets. *Information Processing & Management, 51*(4), 480–499.

Novak, P. K., Smailović, J., Sluban, B., & Mozetič, I. (2015). Sentiment of emojis. *PloS one, 10*(12) E0144296

Nozza, D., Fersini, E., & Messina, E. (2016). *Unsupervised irony detection: A probabilistic model with word embeddings. KDIR*68–76.

Oraby, S., Harrison, V., Misra, A., Riloff, E., & Walker, M. (2017). *Are you serious?: Rhetorical questions and sarcasm in social media dialog. Proceedings of the 18th annual SIGdial meeting on discourse and dialogue*310–319.

Poria, S., Cambria, E., Hazarika, D., & Vij, P. (2016). *A deeper look into sarcastic tweets using deep convolutional neural networks. Proceedings of the 26th international conference on computational linguistics*1601–1612.

Ptáček, T., Habernal, I., & Hong, J. (2014). *Sarcasm detection on Czech and English Twitter. Proceedings of the 25th international conference on computational linguistics*213–223.

Reyes, A., Rosso, P., & Veale, T. (2013). A multidimensional approach for detecting irony in Twitter. *Language Resources and Evaluation, 47*(1), 239–268.

Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., & Huang, R. (2013). *Sarcasm as contrast between a positive sentiment and negative situation. Proceedings of the 2013 conference on empirical methods in natural language processing*704–714.

Rosenthal, S., Farra, N., & Nakov, P. (2017). *Semeval-2017 task 4: Sentiment analysis in Twitter. Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*502–518.

Rosso, P., Rangel, F., Farías, I. H., Cagnina, L., Zaghouani, W., & Charfi, A. (2018). A survey on author profiling, deception, and irony detection for the arabic language. *Language and Linguistics Compass, 12*(4) E12275

Shu, X., Qi, G. J., Tang, J., & Wang, J. (2015). *Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation. Proceedings of the 23rd ACM international conference on multimedia*. ACM35–44.

Sulis, E., Farías, D. I. H., Rosso, P., Patti, V., & Ruffo, G. (2016). Figurative messages and affect in Twitter: Differences between# irony,# sarcasm and# not. *Knowledge-Based Systems, 108*, 132–143.

Tay, Y., Tuan, L. A., Hui, S. C., & Su, J. (2018). *Reasoning with sarcasm by reading in-between. Proceedings of the 56th annual meeting of the association for computational linguistics*.

Wallace, B. C. (2015). Computational irony: A survey and new perspectives. *Artificial Intelligence Review, 43*(4), 467–483.

Wang, Y., Huang, M., Zhao, L., et al. (2016). *Attention-based lstm for aspect-level sentiment classification. Proceedings of the 2016 conference on empirical methods in natural language processing*606–615.

Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big Data, 3*(1), 9.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin, 1*(6), 80–83.

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). *Recognizing contextual polarity in phrase-level sentiment analysis. Proceedings of human language technology conference and conference on empirical methods in natural*347.

Xia, R., Xu, F., Yu, J., Qi, Y., & Cambria, E. (2016). Polarity shift detection, elimination and ensemble: A three-stage model for document-level sentiment analysis. *Information Processing & Management, 52*(1), 36–45.

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., et al. (2016). *Attention-based bidirectional long short-term memory networks for relation classification. Proceedings of the 54th annual meeting of the association for computational linguisticsvol. 2. Proceedings of the 54th annual meeting of the association for computational linguistics* 207–212.