Multi-Modal Sarcasm Detection in Twitter with Hierarchical Fusion Model

Yitao Cai, Huiyu Cai and **Xiaojun Wan**
Institute of Computer Science and Technology, Peking University
The MOE Key Laboratory of Computational Linguistics, Peking University
Center for Data Science, Peking University
{caiyitao,hy_cai,wanxiaojun}@pku.edu.cn

Abstract

Sarcasm is a subtle form of language in which people express the opposite of what is implied. Previous works of sarcasm detection focused on texts. However, more and more social media platforms like Twitter allow users to create multi-modal messages, including texts, images, and videos. It is insufficient to detect sarcasm from multi-model messages based only on texts. In this paper, we focus on multi-modal sarcasm detection for tweets consisting of texts and images in Twitter. We treat text features, image features and image attributes as three modalities and propose a multi-modal hierarchical fusion model to address this task. Our model first extracts image features and attribute features, and then leverages attribute features and bidirectional LSTM network to extract text features. Features of three modalities are then reconstructed and fused into one feature vector for prediction. We create a multi-modal sarcasm detection dataset based on Twitter. Evaluation results on the dataset demonstrate the efficacy of our proposed model and the usefulness of the three modalities.

1 Introduction

Merriam Webster defines sarcasm as *"a mode of satirical wit depending for its effect on bitter, caustic, and often ironic language that is usually directed against an individual"*. It has the magical power to disguise the hostility of the speaker (Dews and Winner, 1995) while enhancing the effect of mockery or humor on the listener. Sarcasm is prevalent on today's social media platforms, and its automatic detection bears great significance in customer service, opinion mining, online harassment detection and all sorts of tasks that require knowledge of people's *real* sentiment.

Twitter has become a focus of sarcasm detection research due to its ample resources of publicly available sarcastic posts. Previous works on Twitter sarcasm detection focus on the text modality and propose many supervised approaches, including conventional machine learning methods with lexical features (Bouazizi and Ohtsuki, 2015; Ptáček et al., 2014), and deep learning methods (Wu et al., 2018; Baziotis et al., 2018).

However, detecting sarcasm with only text modality can never be certain of the true intention of the simple tweet *"What a wonderful weather!"* until the dark clouds in the attached picture (Figure 1(a)) are seen. Images, while are ubiquitous on social platforms, can help reveal (Figure 1(a)), affirm (Figure 1(b)) or disprove the sarcastic nature of tweets, thus are intuitively crucial to Twitter sarcasm detection tasks.

In this work, we propose a multi-modal hierarchical fusion model for detecting sarcasm in Twitter. We leverage three types of features, namely text, image and image attribute features, and fuse them in a novel way. During early fusion, the attribute features are used to initialize a bi-directional LSTM network (Bi-LSTM), which is then used to extract the text features. The three features then undergo representation fusion, where they are transformed into reconstructed representation vectors. A modality fusion layer performs weighted average to the vectors and pumps them to a classification layer to yield the final result. Our results show that all three types of features contribute to the model performance. Furthermore, our fusion strategy successfully refines the representation of each modality and is significantly more effective than simply concatenating the three types of features.

Our main contributions are summarized as follows:

- We propose a novel hierarchical fusion model to address the challenging multi-modal sar-

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2506–2515
Florence, Italy, July 28 - August 2, 2019.

(a) *"What a wonderful weather!"*

(b) *"Yep, totally normal <user>. Nothing is off about this. Nothing at all. #itstoohotalready #climatechangeisreal"*

Figure 1: Examples of image modality aiding sarcasm detection. (a) The image is necessary for the sarcasm to be spotted due to the contradiction of dark clouds in the image and "wonderful weather" in the text; (b) The image affirms the sarcastic nature of the tweet by showing the weather is actually very "hot" and is not at all "totally normal".

casm detection task in Twitter. To the best of our knowledge, we are the first to deeply fuse the three modalities of image, attribute and text, rather than naïve concatenation, for Twitter sarcasm detection.

- We create a new dataset for multi-modal Twitter sarcasm detection and release it[1].
- We quantitatively show the significance of each modality in Twitter sarcasm detection. We further show that to fully unleash the potential of images, we would need to consider image attributes - a high-level abstract information bridging the gap between texts and images.

2 Related Works

2.1 Sarcasm Detection

Various methods have been proposed for sarcasm detection from texts. Earlier methods extract carefully engineered discrete features from texts (Davidov et al., 2010; Riloff et al., 2013; Ptáček et al., 2014; Bouazizi and Ohtsuki, 2015), including n-grams, word's sentiment, punctuations, emoticons, part-of-speech tags, etc. More recently, researchers leverage the powerful techniques of deep learning to get more precise semantic representations of tweet texts. Ghosh and Veale (2016) propose a model with CNN and RNN layers. Besides the tweet content in question, contextual features such as historical behaviors of the author and the audience serve as a good indicator for sarcasm. Bamman and Smith (2015) make use of human-engineered author, audience and response features to promote sarcasm detection. Zhang, Zhang and Fu (2016) concatenate target tweet embeddings(obtained by a Bi-GRU model) with manually engineered contextual features, and show fair improvement compared to completely feature-based systems. Amir et al. (2016) exploit trainable user embeddings to enhance the performance of a CNN classification model. Poria et al. (2016) use the concatenated output of CNNs trained on tweets and pre-trained on emotion, sentiment, personality as the inputs for the final SVM classifier. Y. Tay et al. (2018) come up with a novel multi-dimensional intra-attention mechanism to explicitly model contrast and incongruity. Wu et al. (2018) construct a multi-task model with densely connected LSTM based on embeddings, sentiment features and syntactic features. Baziotis et al. (2018) ensemble a word based bidirectional LSTM and a character based bidirectional LSTM to capture both semantic and syntactic features.

However, little has been revealed by far on how to effectively combine textual and visual information to boost performance of Twitter sarcasm detection. Schifanella et al. (2016) simply concatenate manually designed features or deep learning based features of texts and images to make prediction with two modalities. Different from this work, we propose a hierarchical fusion model to deeply fuse three modalities.

2.2 Other Multi-Modal Tasks

Sentiment analysis is a related task with sarcasm detection. Many researches on multi-modal sen-

[1] https://github.com/headacheboy/data-of-multimodal-sarcasm-detection

timent analysis deal with video data (Wang et al., 2016; Zadeh et al., 2017), where text, image and audio data can usually be aligned and support each other. Though inputs are different, their fusion mechanisms can be inspiring to our task. Poria, Cambria, and Gelbukh (2015) use multiple kernel learning to fuse different modalities. Zadeh et al. (2017) build their fusion layer by outer product instead of simple concatenation in order to get more features. Gu et al. (2018b) align text and audio at word level and apply several attention mechanisms. Gu et al. (2018a) first introduce modality fusion structure attempting to reveal the actual importance of multiple modalities, but their methods are quite different from our hierarchical fusion techniques.

Inspiration can also be drawn from other multimodal tasks, such as visual question answering (VQA) tasks where a frame of image and a query sentence are provided as model inputs. A question-guided attention mechanism is proposed in VQA tasks (Chen et al., 2015) and can boost model performance compared to those using global image features. Attribute prediction layer is introduced (Wu et al., 2016) as a way to incorporate high-level concepts into the CNN-LSTM framework. Wang et al. (2017) exploit a handful of off-the-shelf algorithms, gluing them with a co-attention model and achieve generalizability as well as scalability. Yang et al. (2014) try image emotion extraction tasks with image comments and propose a model to bridge images and comment information by learning Bernoulli parameters.

3 Proposed Hierarchical Fusion Model

Figure 2 shows the architecture of our proposed hierarchical fusion model. In this work, we treat text, image and image attribute as three modalities. Image attribute modality has been shown to boost model performance by adding high-level concept of the image content (Wu et al., 2016). Modality fusion techniques are proposed to make full use of the three modalities. In the following paragraph, we will first define **raw vectors** and **guidance vectors**, and then briefly introduce our hierarchical fusion techniques.

For the image modality, we use a pre-trained and fine-tuned ResNet model to obtain 14×14 regional vectors of the tweet image, which is defined as the raw image vectors, and average them to get our image guidance vector. For the (image) attribute modality, we use another pre-trained and fine-tuned ResNet models to predict 5 attributes for each image, the GloVe embeddings of which are considered as the raw attribute vectors. Our attribute guidance vector is a weighted average of the raw attribute vectors. We use Bi-LSTM to obtain our text vectors. The raw text vectors are the concatenated forward and backward hidden states for each time step of the Bi-LSTM, while the text guidance vector is the average of the above raw vectors. In the belief that the attached image could aid the model's understanding of the tweet text, we apply non-linear transformations on the attribute guidance vector and feed the result to the Bi-LSTM as its initial hidden state. This process is named early fusion. In order to utilize multimodal information to refine representations of all modalities, representation fusion is proposed in which **feature vectors** of the three modalities are reconstructed using raw vectors and guidance vectors. The refined vectors of three modalities are combined into one vector with weighted average instead of simple concatenation in the process of modality fusion. Lastly, the fused vector is pumped into a two layer fully-connected neural network to obtain classification result. More details of our model are provided below.

3.1 Image Feature Representation

We use ResNet-50 V2 (He et al., 2016) to obtain representations of tweet images. We chop the last fully-connected (FC) layer of the pre-trained model and replace it with a new one for the sake of model fine-tuning. Following (Wang et al., 2017), a input image I is re-sized to 448×448 and divided into 14×14 regions. Each region I_i $(i = 1, 2 \ldots, 196)$ is then sent through the ResNet model to obtain a regional feature representation v_{region_i}, a.k.a. a raw image vector.

$$v_{\text{region}_i} = \text{ResNet}(I_i)$$

As is described before, the image guidance vector v_{image} is the average of all regional image vectors.

$$v_{\text{image}} = \frac{\sum_{i=1}^{N_r} v_{\text{region}_i}}{N_r}$$

where N_r is the number of regions and is 196 in this work.

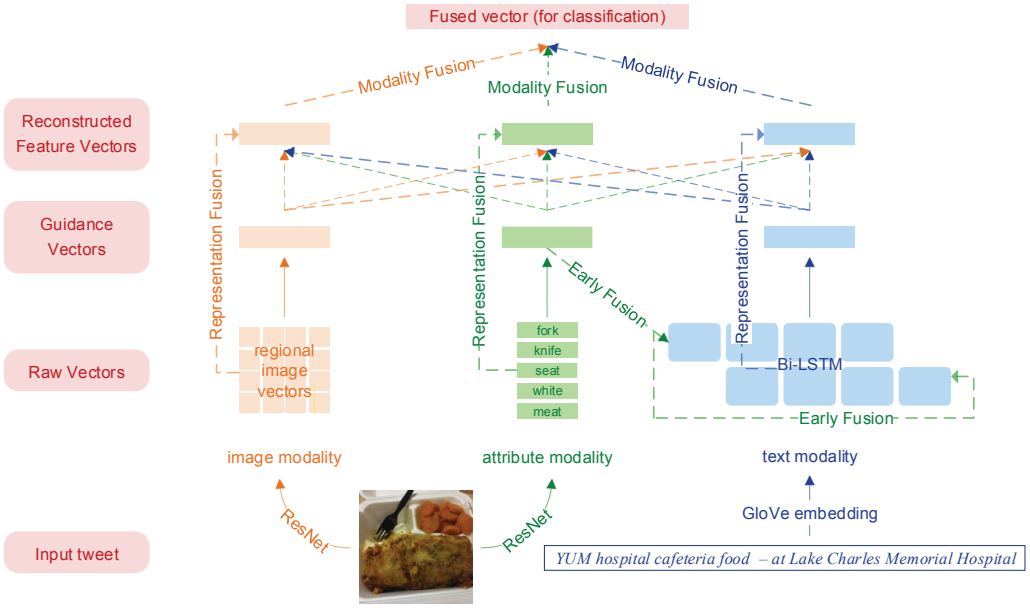

Figure 2: Overview of our proposed model

3.2 Attribute Feature Representation

Previous work (Wu et al., 2016) in image captioning and visual question answering introduces attributes as high-level concepts of images. In their work, single-label and multi-label losses are proposed to train the attribute prediction CNN, whose parameters are transferred to generate the final image representation. While they use parameter sharing for better image representation with attribute-labeling tasks, we take a more explicit approach. We treat attributes as an extra modality bridging the tweet text and image, by directly using the word embeddings of five predicted attributes of each tweet image as the raw attribute vectors.

We first train an attribute predictor with ResNet-101 and COCO image captioning dataset (Lin et al., 2014). We build the multi-label dataset by extracting 1000 attributes from sentences of the COCO dataset. We use a ResNet model pre-trained on ImageNet (Russakovsky et al., 2015) and fine-tune it on the multi-label dataset. Then the attribute predictor is used to predict five attributes a_i $(i = 1, \ldots, 5)$ for each image.

We generate the attribute guidance vector by weighted average. Raw attribute vectors $e(a_i)$ are passed through a two-layer neural network to obtain the attention weights α_i for constructing the attribute guidance vector v_{attr}. The related equations are as follows.

$$\alpha_i = W_2 \cdot \tanh(W_1 \cdot e(a_i) + b_1) + b_2$$
$$\alpha = \text{softmax}(\alpha)$$
$$v_{\text{attr}} = \sum_{i=1}^{N_a} \alpha_i e(a_i)$$

where a_i is the i^{th} image attribute, literally a word out of a vocabulary of 1000; e is the GloVe embedding operation; W_1 and W_2 are weight matrices; b_1 and b_2 are biases; N_a is the number of attributes, and is 5 in our settings.

3.3 Text Feature Representation

Bidirectional LSTM (Bi-LSTM) (Hochreiter and Schmidhuber, 1997) are used to obtain the representation of the tweet text. The equations of operations performed by LSTM at time step t are as follows:

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1})$$
$$f_t = \sigma(W_f \cdot x_t + U_f \cdot h_{t-1})$$
$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1})$$
$$\tilde{c}_t = \tanh(W_c \cdot x_t + U_c \cdot h_{t-1})$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$
$$h_t = o_t \odot \tanh(c_t)$$

where W_i, W_f, W_o, U_i, U_f, U_o are weight matrices; x_t, h_t are input state and hidden state at time step t, respectively; σ is the sigmoid function; $\odot$ denotes element-wise product. The text guidance

vector is the arithmetic average of hidden states in each time step.

$$v_{\text{text}} = \frac{\sum_{i=1}^{L} h_t}{L}$$

where L is the length of the tweet text.

3.4 Early Fusion

The Bi-LSTM initial states are usually set to zeroes in text classification tasks, but it is a potential spot where multi-modal information could be infused to promote the modal's comprehension of the text. In the proposed model, we apply the nonlinearly transformed attribute guidance vector as the initial state of Bi-LSTM.

$$[h_{f0}; h_{b0}; c_{f0}; c_{b0}] = \text{ReLu}(W \cdot v_{\text{attr}} + b)$$

where h_{f0}, c_{f0} are forward LSTM initial states and h_{b0}, c_{b0} are backward LSTM initial states; $[;]$ is vector concatenation; ReLu denotes element-wise application of the Rectified Linear Units activation function; W and b are weight matrix and bias.

We also try to use image guidance vector for early fusion, in which the LSTM initial states are obtained with means similar to the one described above, but it does not perform very well, as will be discussed in the experiments.

3.5 Representation Fusion

Inspired by attention mechanism in VQA tasks, representation fusion aims at reconstructing the feature vectors v_{image}, v_{text}, v_{attr} with the help of low-level raw vectors (namely, the hidden states of time step t $\{h_t\}$ for the text modality, the 196 regional vectors for the image modality, and the five attribute embeddings for the attribute modality) and high-level guidance vectors from different modalities.

We denote $X_m^{(i)}$ as the i^{th} raw vector from modality m (which may be text, image or attribute). The key in this stage is to calculate the weight for each $X_m^{(i)}$. The weighted average then becomes the new representation of modality m.

To leverage as much information as possible and more accurately model the relationship between multiple modalities, we exploit information from all three modalities - more explicitly, guidance vectors v_n where n could be text, image or attribute, when calculating the weights of raw vectors in each modality. For the i^{th} raw vector of each modality m, we calculate three guided weights $\alpha_{mn}^{(i)}$ from the guidance vectors of different modalities n. The final reconstruction weight for the raw vector is the average of the normalized guided weights.

$$\begin{aligned}
\alpha_{mn}^{(i)} &= W_{mn_2} \cdot \tanh(W_{mn_1} \cdot [X_m^{(i)};\ v_n] + b_{mn_1}) \\
&\quad + b_{mn_2} \\
\alpha_{mn} &= \text{softmax}(\alpha_{mn}) \\
\alpha_m^{(i)} &= \frac{\sum_{n \in \{\text{text, image, attr}\}} \alpha_{mn}^{(i)}}{3} \\
v_m &= \sum_{i=1}^{L_m} \alpha_m^{(i)} X_m^{(i)}
\end{aligned}$$

where $m, n \in \{\text{text, image, attr}\}$ denote modalities; $\alpha_{mn}^{(i)}$ is the guided weight for the i^{th} raw vector of modality m under the guidance of modality n, and α_{mn} contains all $\alpha_{mn}^{(i)}$ of all raw vectors of modality m under the guidance of modality n; $\alpha_m^{(i)}$ is the final reconstruction weight for the i^{th} raw vector of modality m; L_m is the length of sequence $\{X_m^{(i)}\}$; W_{mn_1} ,W_{mn_2} are weight matrices and b_{mn_1}, b_{mn_2} are biases.

After representation fusion, v_{image}, v_{text}, v_{attr}, previously denoted as guidance vectors, are now considered feature vectors of each modality and ready to serve as inputs of the next layer.

3.6 Modality Fusion

Instead of simply concatenating the feature vectors from different modalities to form a longer vector, we perform modality fusion motivated by the work of (Gu et al., 2018a). The feature vector for each modality m, denoted as v_m, is first transformed into a fixed-length form v'_m. A two-layer feed-forward neural network is implemented to calculate the attention weights for each modality m, which is then used in the weighted average of transformed feature vectors v'_m. The result is a single, fixed-length vector v_{fused}.

$$\begin{aligned}
\tilde{\alpha}_m &= W_{m_2} \cdot \tanh(W_{m_1} \cdot v_{\text{m}} + b_{m_1}) + b_{m_2} \\
\tilde{\alpha} &= \text{softmax}(\tilde{\alpha}) \\
v'_m &= \tanh(W_{m_3} \cdot v_m + b_{m_3}) \\
v_{\text{fused}} &= \sum_{m \in \{\text{text, image, attr}\}} \tilde{\alpha}_m v'_m
\end{aligned}$$

where m is one of the three modalities and $\tilde{\alpha}$ is a vector containing $\tilde{\alpha}_m$; W_{m_1}, W_{m_2}, W_{m_3} are

	Training	Development	Test
sentences	19816	2410	2409
positive	8642	959	959
negative	11174	1451	1450

Table 1: Statistics of our dataset

Hyper-parameters	Value
LSTM hidden size	256
Batch size	32
Learning rate	0.001
Gradient Clipping	5
Early stop patience	5
Word and attribute embedding size	200
ResNet FC size	1024
Modality fusion size	512
LSTM dropout rate	0.2
Classification layer l2 parameters	1e-7

Table 2: Hyper-parameters

weight matrices. $b_{m_1}, b_{m_2}, b_{m_3}$ are biases; v_{m} represents reconstructed feature vectors in the representation fusion process.

3.7 Classification layer

We use a two layer fully-connected neural network as our classification layer. The activation function of the hidden layer and the output layer are element-wise ReLu and sigmoid functions, respectively. The loss function is cross entropy.

4 Dataset and Preprocessing

There is no publicly available dataset for evaluating the multi-modal sarcasm detection task, and thus we build our own dataset, which will be released later. We collect and preprocess our data similar to (Schifanella et al., 2016). We collect English tweets containing a picture and some special hashtag (e.g., *#sarcasm*, etc.) as positive examples (i.e. sarcastic) and collect English tweets with images but without such hashtags as negative examples (i.e. not sarcastic). We further clean up the data as follows. First, we discard tweets containing *sarcasm*, *sarcastic, irony, ironic* as regular words. We also discard tweets containing URLs in order to avoid introducing additional information. Furthermore, we discard tweets with words that frequently co-occur with sarcastic tweets and thus may express sarcasm, for instance *jokes*, *humor* and *exgag*. We divide the data into training set, development set and test set with a ratio of 80%:10%:10%. In order to evaluate models more accurately, we manually check the development set and the test set to ensure the accuracy of the labels. The statistics of our final dataset are listed in table 1.

For preprocessing, we first replace mentions with a certain symbol $\langle user \rangle$. We then separate words, emoticons and hashtags with the NLTK toolkit. We also separate hashtag sign # from hashtags and replace capitals with their lowercases. Finally, words appearing only once in the training set and words not appearing in the training set but appearing in the development set or test set are replaced with a certain symbol $\langle unk \rangle$.

5 Experiments

5.1 Training Details

Pre-trained models. The pre-trained ResNet model is available online. The word embeddings and attribute embeddings are trained on the Twitter dataset using Glove (Pennington et al., 2014).
Fine tuning. Parameters of the pre-trained ResNet model are fixed during training. Parameters of word and attribute embeddings are updated during training.
Optimization. We use the Adam optimizer (Kingma and Ba, 2014) to optimize the loss function.
Hyper-parameters. The hidden layer size in the neural networks described in the fusion techniques is half of its input size. Other hyper-parameters are listed in table 2.

5.2 Comparison Results

Table 3 shows the comparison results (F-score and Accuracy) of baseline models and our proposed model. We implement models with one or multiple modalities as baseline models. We also present the results of naïve solution (all negative, random) of this task.
Random. It randomly predicts whether a tweet is sarcastic or not.
Text(Bi-LSTM). Bi-LSTM is one of the most popular method for addressing many text classification problems. It leverages a bidirectional LSTM network for learning text representations and then uses a classification layer to make prediction.
Text(CNN). CNN is also one of the state-of-the-art methods to address text classification problems. We implement text CNN (Kim, 2014) as a baseline model.

Model	F-score	Pre	Rec	Acc
All negative	-	-	-	0.6019
Random	0.4470	0.4005	0.5057	0.5027
Text(Bi-LSTM)	0.7753	0.7666	0.7842	0.8190
Text(CNN)	0.7532	0.7429	0.7639	0.8003
Image	0.6153	0.5441	0.7080	0.6476
Attr	0.6334	0.5606	0.7278	0.6646
Concat(2)	0.7799	0.7388	0.8259	0.8103
Concat(3)	0.7874	0.7336	0.8498	0.8174
Our model	**0.8018**	0.7657	0.8415	**0.8344**

Table 3: Comparison results

Image. Image vectors after the pooling layer of ResNet are inputs of the classification layer. We only update parameters of the classification layer.
Attr. Since image attribute is one of the modalities in our proposed model, we also try to use only attribute features to make prediction. The attribute feature vectors are inputs of the classification layer.
Concat. Previous work (Schifanella et al., 2016) concatenates different feature vectors of different modalities as the input of the classification layer. We implement this concatenation model with our feature vectors of different modalities and apply it for classification. The number in parentheses is the number of modalities we use. (2) means concatenating text features and image features, while (3) means concatenating all text, image and attribute features.

We can see that the models based only on the image or attribute modality do not perform well, while Text(Bi-LSTM) and Text(CNN) models perform much better, indicating the important role of text modality. The Concat(3) model outperforms Concat(2), because adding attributes as a new modality actually introduces external semantic information of images and helps the model when it fails to extract valid image features. Our proposed hierarchical fusion model further improves the performance and achieves the state-of-the-art scores, revealing that our fusion model leverages features of three modalities in a more effective way.

We further apply sign tests between our proposed model and Text(Bi-LSTM), Concat(2), Concat(3) models. The null hypotheses are that our proposed model doesn't perform better than each baseline model. The statistics of the sign tests are listed in table 4. All significance levels are less than 0.05. Therefore, all of the null hypotheses is rejected and our proposed model significantly performs better than baseline models.

	Concat(3)	Concat(2)	Text(Bi-LSTM)
t^+	106	149	120
t^-	65	91	83
p	0.0011	0.0001	0.0057

Table 4: Statistics of sign tests. (t^+ is the number of tweets that our proposed model predicts them right but baseline models do not. t^- is the number of tweets that baseline models predict them right but our proposed model does not. p is the significance value.)

5.3 Component Analysis of Our Model

We further evaluate the influence of early fusion, representation fusion, as well as different modality representation in early fusion on the final performance. The evaluation results are listed in Table 5.

	F-score	Pre	Rec	Acc
w/o EF	0.7880	0.7570	0.8217	0.8240
w/o RF	0.7902	0.7456	0.8405	0.8223
EF(img)	0.7787	0.7099	0.8624	0.8049
Our model	**0.8018**	0.7657	0.8415	**0.8344**

Table 5: Ablation study. 'w/o' means removal of this component. EF denotes early fusion. RF denotes representation fusion. EF(img) means using image guidance vectors for early fusion.

We can see that the removal of early fusion decreases the performance, which shows that early fusion can improve the text representation. Early fusion with attribute representation performs better than that with image representation, indicating the gap between text representation and image representation. If representation fusion is removed, the performance is also decreased, which indicates that representation fusion is necessary and that the representation fusion can refine the feature representation of each modality.

6 Visualization Analysis

6.1 Running Examples

Figure 3 shows some sarcastic examples that our proposed model predicts them correctly while the model with only text modality fails to label them right. It shows that with our model, images and attributes can contribute to sarcasm detection. For example, an image with a dangerous tackle and a text saying *'not dangerous'* convey strong sarcasm in example (a). *'Respectful customers'* is contradicted to the messy parcels as well as the attribute

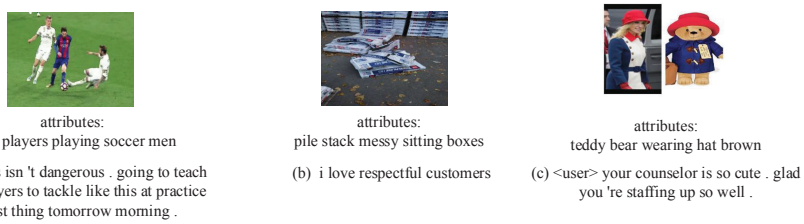

Figure 3: Examples of sarcastic tweets

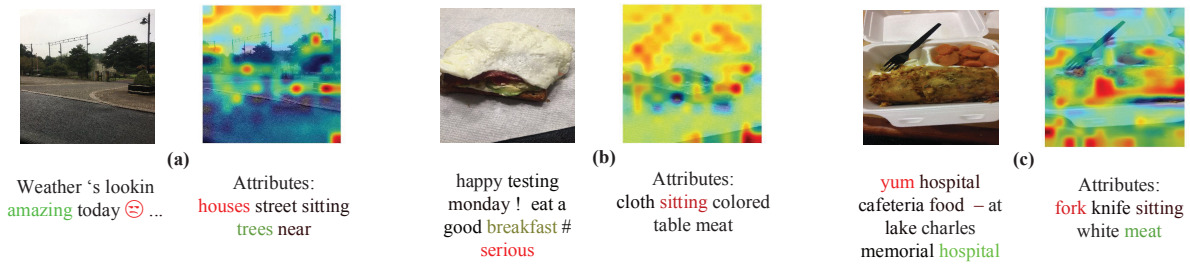

Figure 4: Attention visualization of sarcastic tweets

'messy' in example (b). Without images, successfully detecting these sarcasm instances is almost impossible. The model with only text modality fails to detect sarcasm as for example (c), though the word *so* is repeated several times in example (c). However, with image and attribute modalities, our proposed model correctly detects sarcasm in these tweets.

6.2 Attention Visualization

Figure 4 shows the attention of some examples at the representation fusion stage. Our model can successfully focus on the appropriate parts of the image, the essential words in the sentences and the important attributes. For example, our model pays more attention on the unamused face emoji and the word *'amazing'* for texts, and pays more attention on the gloomy sky in example (a), thus this tweet is predicted as sarcastic tweet because of the inconsistency of these two modalities. In example (b), our model focuses on the word *'serious'* in texts and focuses on the simple meal in the picture that contradicts to the *'good breakfast'*, revealing that this tweet should be sarcastic. In example (c), the word *'yum'*, the attribute *'meat'* and the food in the image indicate the sarcastic meaning of the tweet.

6.3 Error Analysis

Figure 5 shows an example that our model fails to label it right.

Figure 5: Example of misclassified samples

In the example, the insulting gesture in the picture is contrast to the phrase *'thanks for'*. However, the model is unable to obtain the common sense that this gesture is insulting. Therefore, the attention of this picture does not focus on the insulting gesture. Moreover, attributes do not reveal the insulting meaning of the pictures as well, thus our model fails to predict this tweet as sarcastic.

7 Conclusion and Future Work

In this paper we propose a new hierarchical fusion model to make full use of three modalities (images, texts and image attributes) to address the challenging multi-modal sarcasm detection task. Evaluation results demonstrate the effectiveness of our proposed model and the usefulness of the three modalities. In future work, we will incorporate other modality such as audio into the sarcasm detection task and we will also investigate to make use of common sense knowledge in our model.

Acknowledgment

This work was supported by National Natural Science Foundation of China (61772036) and Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology). We thank the anonymous reviewers for their helpful comments. Xiaojun Wan is the corresponding author.

References

Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. *CoRR*, abs/1607.00976.

David Bamman and Noah A. Smith. 2015. Contextualized sarcasm detection on twitter. In *ICWSM*.

Christos Baziotis, Nikos Athanasiou, Pinelopi Papalampidi, Athanasia Kolovou, Georgios Paraskevopoulos, Nikolaos Ellinas, and Alexandros Potamianos. 2018. Ntua-slp at semeval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive rnns. *arXiv preprint arXiv:1804.06659*.

M. Bouazizi and T. Ohtsuki. 2015. Sarcasm detection in twitter: "all your products are incredibly amazing!!!" - are they really? In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6.

Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. 2015. Abc-cnn: An attention based convolutional neural network for visual question answering. *arXiv preprint arXiv:1511.05960*.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, CoNLL '10, pages 107–116, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shelly Dews and Ellen Winner. 1995. Muting the meaning a social function of irony. *Metaphor and Symbol*, 10(1):3–19.

Aniruddha Ghosh and Dr. Tony Veale. 2016. Fracking sarcasm using neural network. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 161–169. Association for Computational Linguistics.

Yue Gu, Kangning Yang, Shiyu Fu, Shuhong Chen, Xinyu Li, and Ivan Marsic. 2018a. Hybrid attention based multimodal network for spoken language classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2379–2390. Association for Computational Linguistics.

Yue Gu, Kangning Yang, Shiyu Fu, Shuhong Chen, Xinyu Li, and Ivan Marsic. 2018b. Multimodal affective analysis using hierarchical attention strategy with word-level alignment. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2225–2235. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. *CoRR*, abs/1603.05027.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2015. Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2539–2544.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. 2016. A deeper look into sarcastic tweets using deep convolutional neural networks. *CoRR*, abs/1610.08815.

Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223. Dublin City University and Association for Computational Linguistics.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *EMNLP 2013 - 2013*

Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, pages 704–714. Association for Computational Linguistics (ACL).

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Rossano Schifanella, Paloma de Juan, Joel R. Tetreault, and Liangliang Cao. 2016. Detecting sarcasm in multimodal social platforms. *CoRR*, abs/1608.02289.

Yi Tay, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. 2018. Reasoning with sarcasm by reading in-between. *CoRR*, abs/1805.02856.

Haohan Wang, Aaksha Meghawat, Louis-Philippe Morency, and Eric P. Xing. 2016. Select-additive learning: Improving cross-individual generalization in multimodal sentiment analysis. *CoRR*, abs/1609.05244.

Peng Wang, Qi Wu, Chunhua Shen, and Anton van den Hengel. 2017. The vqa-machine: Learning how to use existing vision algorithms to answer new questions. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn*, volume 4.

Chuhan Wu, Fangzhao Wu, Sixing Wu, Junxin Liu, Zhigang Yuan, and Yongfeng Huang. 2018. Thu_ngn at semeval-2018 task 3: Tweet irony detection with densely connected lstm and multi-task learning. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 51–56.

Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony Dick, and Anton van den Hengel. 2016. What value do explicit high level concepts have in vision to language problems? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 203–212.

Yang Yang, Jia Jia, Shumei Zhang, Boya Wu, Qicong Chen, Juanzi Li, Chunxiao Xing, and Jie Tang. 2014. How do your friends on social media disclose your emotions? In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 306–312.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor fusion network for multimodal sentiment analysis. *CoRR*, abs/1707.07250.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Tweet sarcasm detection using deep neural network. In *COLING*.