# In Your Face: Sentiment Analysis of Metaphor with Facial Expressive Features

Dongyu Zhang
*School of Software*
*Dalian University of Technology*
Dalian 116620, China
zhangdongyu@dlut.edu.cn

Minghao Zhang
*School of Software*
*Dalian University of Technology*
Dalian 116620, China
zhang.minghao@outlook.com

Teng Guo
*School of Software*
*Dalian University of Technology*
Dalian 116620, China
teng.guo@outlook.com

Ciyuan Peng
*School of Engineering, IT and Physical Sciences*
*Federation University Australia*
Ballarat, VIC 3353, Australia
sayeon1995@gmail.com

Vidya Saikrishna
*School of Engineering, IT and Physical Sciences*
*Federation University Australia*
Ballarat, VIC 3353, Australia
v.saikrishna@federation.edu.au

Feng Xia
*School of Engineering, IT and Physical Sciences*
*Federation University Australia*
Ballarat, VIC 3353, Australia
f.xia@ieee.org

***Abstract*—Metaphor plays an important role in human communication, which often conveys and evokes sentiments. Numerous approaches to sentiment analysis of metaphors have thus gained attention in natural language processing (NLP). The primary focus of these approaches is on linguistic features and text rather than other modal information and data. However, visual features such as facial expressions also play an important role in expressing sentiments. In this paper, we present a novel neural network approach to sentiment analysis of metaphorical expressions that combines both linguistic and visual features and refer to it as the multimodal model approach. For this, we create a Chinese dataset, containing textual data from metaphorical sentences along with visual data on synchronized facial images. The experimental results indicate that our multimodal model outperforms several other linguistic and visual models, and also outperforms the state-of-the-art methods. The contribution is realized in terms of novelty of the approach and creation of a new, sizeable, and scarce dataset with linguistic and synchronized facial expressive image data. The dataset is particularly useful in languages other than English and the approach addresses one of the most challenging NLP issue: sentiment analysis in metaphor.**

***Index Terms*—Sentiment Analysis, Metaphor Identification, Facial Images, Linguistics Features, Multimodal Model**

## I. Introduction

Metaphor is pervasive in human language, and it is important in conceptual knowledge [1]–[3]. In metaphorical language, humans use one concept, typically physical, concrete, and simple, to express another, typically abstract, vague, and complex concept [4], [5]. For example, "time is money." Time is metaphorically viewed as money to emphasize that it is valuable. In another instance, "She killed my fear," my fear is described as a living thing, and thus, dispelling it relates to killing. Metaphor involves mapping between two domains and conceptualizing one domain (target) in terms of another (source).

Sentiment or emotion, as a widely used abstract conception, is frequently communicated and conceptualized by metaphors [6], [7]. Generally, two different metaphors convey and evoke sentiment. In one case, the target domain is sentiment itself. For instance, in "she was boiling at what he did," the angry, emotional self is conceptualized as steam, and so it is expressed metaphorically in terms of boiling. Other metaphors not about emotion, but they have sentiment or emotional connotations. For example, in the metaphorical stance "the inflation has eaten up him," the target domain is inflation and the source domain, implied by the verb eaten up, is some kind of fierce beast. This metaphorical instance may thus express senses of fear and negative sentiment about inflation. Neuroscience research suggests that metaphorical texts are associated more with the activation in the amygdala that process emotion than with literal areas [8].

Scholars in artificial intelligence [9] and natural language processing (NLP) have realised the importance of metaphors in expressing sentiments. Early research works in sentiment analysis in metaphors focussed on computational methods that aimed at performing sentiment analysis based on linguistic features and text resources, thus completely ignoring other modal information and data. But it is evident from the research work carried out by [10]–[13] that humans often express and convey sentiments in multimodal ways i.e. combination of textual, visual and audio ways. Using a combination of different ways for sentiment analysis have proven to be successful in the work carried out by [14], [15] and, we therefore consider visual features such as facial expressions in our research work for metaphor sentiment analysis. Visual features such as facial expressions strongly relate to emotional state and therefore considered very important for emotional communication and detection [16], [17].

We therefore propose a novel approach that combines both textual and facial representations for sentiment analysis of metaphors. To validate our approach, we constructed a dataset containing Chinese metaphorical textual instances with manual

annotations of sentiment accompanying facial images of individuals reading metaphorical sentences. We then used facial expression and linguistic features to analyse sentiments of metaphor. We employed a feature fusion approach to combine textual and visual features that improve the performance in terms of classification of sentiments of metaphor. Experimental results demonstrate that our multimodal model outperforms textual and visual models separately. Our approach significantly outperforms the state-of-the-art method. In short, the contributions are as listed below.

- A novel neural network approach is proposed for sentiment analysis of metaphorical expressions, which explores linguistic and facial expressive features.
- A novel and scarce dataset is presented, which will be released publicly with linguistic and synchronized facial expressive image data. The dataset is particularly useful for Chinese language processing.
- Experimental results prove the efficiency of our approach in terms of classification as compared to other textual and facial models separately.
- Our findings add psychological evidence on the relationship between sentiment and metaphor.

The rest of the article is organised as follows. Section II reviews related work that presents a couple of techniques and datasets in relation to sentiment analysis of metaphors. Section III describes the data used in the experiments in detail. Section IV explains the methodology proposed. Section V shows experimental results. Analysis is carried out on the experimental results and its efficiency and comparison is made with state of art methods in the same section. We finally conclude and discuss future research in Section VI.

## II. Related Work

This section discusses datasets and methodologies in similar research works to ours. Section 2.1 talks about the existing datasets and Section 2.2 explains existing methodologies used in metaphor sentiment analysis.

### A. Datasets

Ghosh et al. [18] proposed a figurative language dataset collected from Twitter. The types considered were sarcasm, irony, metaphor, and others. They collected content and hashtags for each tweet, and annotated the sentiment score of the tweets on an 11-point scale, ranging from -5 (very negative) to 5 (very positive), with 0 meaning neutral. The dataset was then used in metaphor sentiment analysis. Another dataset in Kozareva's study [19] was based on annotated data from [20]. Kozareva collected metaphors in English, Spanish, Russian, and Farsi in the governance domain. This dataset provided the sentiment polarity and valence scores for the sentiment analysis experiment.

There are a few datasets known to be useful for sentiment analysis of Chinese metaphors. Peng et al. [21] used a manually annotated dataset which had a collection of 4,900 metaphorical contexts including 2,168 negative samples and 2,732 positive ones from Chinese literature. They annotated the target and sentiment of the metaphors with three native Chinese speakers. Lu et al. [22] built a multilingual annotated corpus containing 5,422 tweets on four topics (iPhone 6, Windows 8, Vladimir Putin, and Scottish Independence) in English, Japanese, and Chinese. They provided various information on Twitter, including emotional signals with polarity, degree modifiers, subtopics, hashtags, global sentiment polarity, and rhetorical devices (metaphor, comparison, sarcasm, rhetorical question, and non-rhetoric). Unlike previous studies focusing on textual data, this paper involves textual data from metaphorical expressions along with visual data on synchronized facial images.

### B. Automatic Emotion Recognition in Metaphor

It has been proven that metaphors contain more intense emotions than literal expressions [23]. Therefore, in daily life, people usually use metaphors to express their emotions. The researchers, nowadays, are shifting their focus to the study of sentiment analysis in metaphors. Numerous research works have been carried out in this direction leading to a number of metaphorical sentiment detection methods.

Strzalkowski et al. [24] proposed a rule-based metaphorical classification model. The authors built a calculus based on an extended version of the Affective Norms in English Words (ANEW) psycholinguistic database. They also extended and improved this calculus to capture prior affect brought by metaphor's direct context. Nguyen et al. [25] used a statistical approach named Figurative Language Analysis. This model could classify tweets into three categories by term features and emotional patterns.

Some researchers used machine learning methods [26], [27] to classify the polarities of metaphorical emotions and used regression models to score emotions. Kozareva [19] combined the triggering of cognition, emotion, perception, and social processes with stylistic and lexical information. By analysing English, Spanish, Russian, and Persian datasets, Kozareva showed that the development of implicitly rich texts affects polarity and value prediction techniques that are portable between languages. McGillion et al. [28] proposed a stacking system composed of several regression models (including ridge regression, mixed Gaussian model, and Bayesian ridge regression) to predict sentiment score on figurative language from Twitter. For the same task, Patra et al. [29] used textual and sentiment features, including parts of speech, sentiment features, intensifiers, and sentiment abruptness, to train a multilevel classification model to improve the prediction performance.

Rentoumi et al. [30] proposed a method for sentiment analysis of metaphorical language, using word sense disambiguation, giving polarity to the meaning of the word using an n-gram-based method. The polarity of the sense is combined with the context price converter. The hidden Markov model further assigns polarity to the sentence. Zhang et al. [31] contributed a deep learning method based on an attention-based long short-term memory (LSTM) network. The model could detect the binary sentiment of metaphorical context effectively.

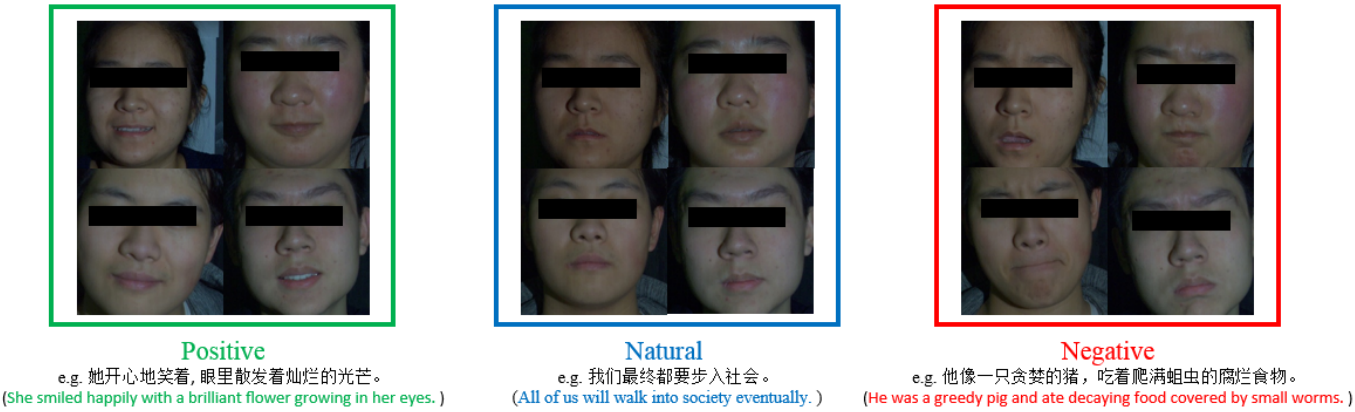


Fig. 1. Examples of metaphorical sentences with positive, neutral and negative sentiments.

It described the interaction using attention mechanisms and standard LSTM. Dankers et al. [32] used a multi-task learning method to jointly learn metaphors and metaphorical emotions through hard parameter sharing and soft parameter sharing. Similarly, Huguet et al. [33] proposed the first combined model of metaphor, emotion, and political rhetoric, and proved that they can improve performance in the three tasks. There were works related to the use of support vector machines in the recognition of facial expressions in references [34] [35] [36].

## III. Dataset

### A. Textual Data

The textual data used in our experiments originated from the works of Zhang et al. [37]. It consists of a Chinese metaphor corpus with 5,605 metaphorical sentences containing annotations of sentiments. To build this corpus, Zhang *et al.* collected real-world Chinese sentences with abundant emotional information from different sources, including books, journals, movie scripts and networks. In this corpus, there are three categories of manually annotated sentiment: positive, negative and neutral. The emotion intensity has five degrees (1, 3, 5, 7, and 9) for both positive and negative sentiments, and one for neutral sentiment (0). We selected sentences with an average length of 15.5 Chinese characters. The emotion intensity of all selected sentences for both positive and negative sentiments were all equal to or greater than 7. This eventually meant that the sentiments in these sentences were strong and clear. The selection process resulted in 725 sentences with negative sentiments, 491 sentences with positive sentiments and 1005 sentences with neutral sentiment. We used them in the process of visual data collection and textual feature extraction. Examples of metaphorical sentences with positive, neutral and negative sentiments are given in Figure 1.

### B. Visual Data

To collect visual data, online and offline campus advertisements were posted to gather student participation in this research work. All the participants were first asked to provide consent to take part in the research work. Participants were also requested to provide their written consent to taking photographs and other relevant data that was going to be used in current experiment and possibly in future experiments as well. A total of 40 participants (27 males and 13 females) took part in the survey. They were aged between 18-21. Each participant was randomly allocated 125 sentences from the dataset which had combination of all three types of sentiments (positive, negative and neutral). Every sentence was read by randomly selected 5 other students to minimize the error brought by the different participants. We followed the below procedure for data collection.

- Each participant sat at a distance of 90 cms from the scanner. They were asked to remove glasses (if they wore any) and asked not to move.
- The sentences were displayed on screen one by one.
- Each sentence was displayed for a period of 10 seconds to know the sentiment.
- The scanner collected the facial expression change 1-2 seconds after the sentences were read.
- This process resulted in 50-130 colored photos per student per sentence.

Using sentence embedding alone to classify sentiment in metaphors is difficult and less efficient. To improve the efficiency (which means improving performance by correct classification), we added visual features extracted from photos of people reading sentences. For this we used two 3-D scanners [38] which were set up using two Basler cameras (acA640-750um and acA1300-200uc) and a digital light projector (LightCrafter 4500 EVM) to screen high-definition face images. This scanner has the capacity to output a series of point clouds (or RGB-D data) and color textures at 120 fps, so that subtle facial expression changes can be captured.

Using the above process, 327,793 images of human faces were generated in total. Each sentence had around 328 images that reflected the sentiment in the form of human expressions. The original image size (1280 $\times$ 1024) which had background, was reduced to a size of (300 $\times$ 300) by removing the background in the preprocessing step, as can be seen in Figure 2. The feature points on these images were extracted by a

classical face recognition algorithms [39].

## IV. Method

In this paper, we used a fusion model to predict the sentiment of metaphors. The structure of this model is shown in Figure 3. The entire methodology was executed in three steps. In the first step, an LSTM network was used with attention to learn the linguistic embedding of sentences [40]. In the second step, we added the embedding of matching facial expressions learned by convolutional neural networks (CNNs) [41], [42]. In the third step, three kinds of fusion methods were used to combine the linguistic and visual features. Each one of the steps are explained in detail in the sections below.

### A. Learning Linguistic Representations

Peng et al. [21] found that using an attention mechanism to capture the importance of each word in the LSTM network can improve the sentiment classification performance. We therefore trained an LSTM-based model with attention mechanism to classify sentiment of metaphors. The model was used to learn representation of sentences in test dataset and generated vectors. The vectors were then used to classify sentiments of these sentences that were to be used later in the experiments.

We implemented the model primarily using Python tool package, Keras[1]. An embedding layer was used to load the word vectors that were pre-trained on the microblog corpus of every sentence, $S = (w_1, w_2, \ldots, w_n)$, where $w_i$ is the word embedding vector. They were then fed into the LSTM layers that generated sequences of hidden states. Let $\tilde{x_t} = [h_{t-1}, x_t]$; each cell in the LSTM layer performs the following computation:

$$f_t = \sigma(W_f \cdot \tilde{x_t} + b_f), \tag{1}$$

$$i_t = \sigma(W_i \cdot \tilde{x_t} + b_i), \tag{2}$$

$$\tilde{C_t} = \sigma(W_C \cdot \tilde{x_t} + b_C), \tag{3}$$

where $\sigma$ is the sigmoid function, $h_{t-1}$ is the output of the last cell, and $x_t$ is the input word vector of the current cell. $W_f$, $W_i$, $W_C$ and $b_f$, $b_i$, $b_C$ are the weight matrices and biases of the corresponding layers in an LSTM cell.

Then, the cell state is updated by

$$C_t = f_t * C_{t-1} + i_t * \tilde{C_t}. \tag{4}$$

The output of current cell is generated by

$$h_t = \sigma(W_o \cdot \tilde{x_t} + b_o) * tanh(C_t), \tag{5}$$

where $W_o$ and $b_o$ are the weight matrix and bias of the output gate in the current cell of the LSTM layer respectively.

We then put these sequences of hidden states into the attention layer, which could capture the importance of each word. The weight of every word in a sentence was calculated by:

$$A(h_i) = tanh((h_i * W)^T + b), \tag{6}$$

[1]https://github.com/keras-team/keras

Fig. 2. Examples of images of facial expressions.

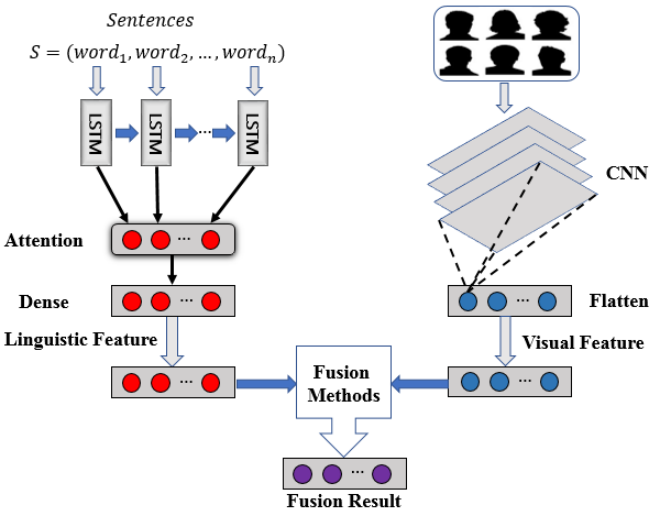


Fig. 3. The process of linguistic and visual feature extraction and fusion.

where $h_i$ is the output of the $i$th word in the sentence of the LSTM layer. $W$ and $b$ are the weight matrix and the bias of the LSTM layer respectively. Finally, we generate the input vector $x$ of the next layer using the weighting summation of the output vectors of the LSTM layer:

$$x = \sum_{i=1}^{N} A(h_i) * h_i. \tag{7}$$

In our experiments, we used 300 dimensions of word vectors as the input. We set the output of the LSTM layer to 100 dimensions. The length of sentences were set to 10 words. If there were less than 10 words in a sentence, the word embedding sequences of the sentences were filled with 300 dimension vectors consisting of 0s. A dense layer with a dropout rate of 0.2 was also added before the output layer to avoid overfitting.

The output layer used a softmax activation function. We optimized the model using the *RMSProp* method, and the loss

function was *binary crossentropy*. Moreover, the model used the backpropagation algorithm to reduce the loss rate for the whole training process of the model.

We randomly selected sentences from our dataset in Section III-A to train this model. This contained 1,221 sentences in the training set and 1,000 sentences in the testing set (Table I). Firstly, we used a Chinese word segmentation tool called jieba[2] to divide each sentence into words. The rationale behind using the segementation tool jieba is that, it is commonly used tool that works on word frequency statistics and considered very light-weight and fast. We filtered out the stop words using jieba.

TABLE I
THE NUMBER OF SAMPLES IN THE TRAIN AND TEST TEXTUAL DATASETS.

| Sentiment Type | Train | Test |
|---|---|---|
| Natural | 505 | 500 |
| Negative | 385 | 340 |
| Positive | 331 | 160 |

The model was trained using a 10-fold cross-validation technique on the training dataset to obtain super parameters. The trained model resulted in a penultimate layer that represented sentence with a 100-dimensional vector. This vector was going to be used as features in the feature fusion experiment. Testing was performed on the test dataset and the results are discussed in Section V.

### B. Learning Visual Representations

The features extracted by Dlib (Python) from the facial expressions cannot detect emotions. Therefore, all the images after preprocessing are fed into a CNN model to extract the corresponding visual representations. The structure of the model is shown in Figure 4. A 100-dimension vector representation of the image is finally obtained.

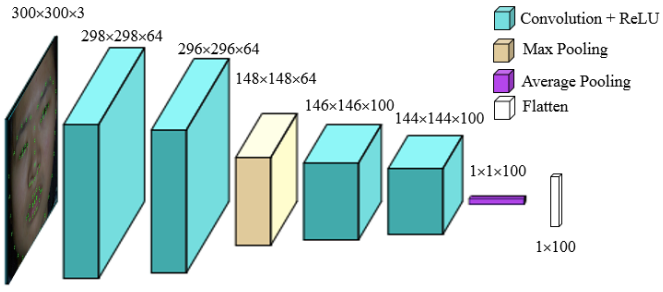


Fig. 4. The CNN model to extract features of facial expressions.

The network shown in Figure 4 has two blocks and seven layers. Each block has two convolution layers with $3 \times 3$ kernels and a pooling layer. The output size of the convolution layer was reduced because of valid padding. We used ReLU as the activation function of the convolution layers. We used a max pooling in the first block with $2 \times 2$ kernels and an average pooling in the second block with $144 \times 144$ kernels. Finally, we flattened the output of these blocks into a $1 \times 100$ dimensional vector.

### C. Fusion

This section details the process of blending linguistic and visual features. Two methods are proposed for processing visual image data, and five visual feature vectors are obtained by these two methods. Finally, the text features are merged with the visual features by using three fusion methods.

**Method of processing visual image data**. We collected five participants' facial expressions for each sentence. We captured each participant's expression changes in a sequence of images, resulting in hundreds of pictures for each sentence.

The captured expressions of participants as they were reading the metaphors, were divided into three stages: pre-change, changing, and post-change. The information in pre-change and post-change expressions were little useful, whereas the information in changing expression was most obvious and useful to be considered for vector embedding and represented as image representation vectors. To obtain suitable visual image feature vectors, two methods were used to process image data: the K-means clustering method and the intermediate value method.

We used a clustering algorithm to sort the image features into three categories matching the change stages, and we used the cluster center of each class as a feature vector to represent student expression changes. We used the K-means model in scikit-learn[3], an easy and useful Python toolkit, to complete the experiment. We set $K = 3$ to match the number of categories.

The second preprocessing method was to select the intermediate images. We assumed that pictures showing changes in expression were in the middle of the sequences. We selected three images from each student's photo sentence, and used the average of these vectors as the student's facial expression change. Then, we used the average of five students' representation vectors as visual features of sentences. We also tried to use the average of vectors of the middle 20% of every photo sequence as representation vectors of the students' expression changes.

Using the two methods introduced above, we obtained five kinds of feature vectors to represent facial expression changes after reading every sentence. In the feature fusion step, we tried three methods to fuse them with the linguist vectors.

**Fusion method**. We used three methods to fuse image features with linguistic feature vectors: feature fusion addition (FFA), feature fusion concatenation (FFC), and prediction result fusion (RF).

In the FFA method, we added the visual image feature vector to the linguistic feature vector to obtain a 100-dimensional fusion vector. Then we used this vector to train a linear regression (LR) classification model. The method was:

$$V_F = V_L + V_I, \tag{8}$$

where $V_F$ is the fusion feature vector, $V_L$ is the linguistic feature vector, and $V_I$ is the visual image feature vector.

[2]https://github.com/fxsjy/jieba

[3]https://scikit-learn.org/stable/

In the FFC method, we concatenated the visual image feature vector with the linguistic feature vector to obtain a 200-dimensional fusion vector. We used LR for prediction. The method was:

$$V_F = concatenate(V_L, V_I), \tag{9}$$

where $V_F$ is the fusion feature vector, $V_L$ is the linguistic feature vector, and $V_I$ is the visual image feature vector.

In the RF method, like [43], we first used the linguistic and visual embedding vectors to train the LR classification model respectively, and we obtained the probability of all test samples belonging to every class. We then added these two probability vectors with weights. For all test samples, we selected the class with the largest probability as the result of the fusion prediction, which was:

$$C_F^* = \arg\max_{C_F}[P_{LR}(V_L) \cdot i + P_{LR}(V_I) \cdot j], \tag{10}$$

where $C_F^*$ is the result of the fusion prediction: positive ($P$), neutral ($O$), and negative ($N$), $P_{LR}(V_L)$ is the prediction result using the linguistic feature vector and $P_{LR}(V_I)$ is the prediction result using the visual image feature vector. $i$ is the weight (0 to 1) and $j = 1 - i$. The result was the class that maximized the prediction probability.

## V. Results and Discussion

We first tested performance of only using visual features (the first, second, and third cluster center [CC 1, CC 2 and CC 3 in tables], and the average vectors of features of 3 or 20% [Mid 3 and Mid 20% in tables]) to predict sentiment of sentences and chose the best kind of visual features.We tried all fusion methods (feature fusion by adding, feature fusion by concatenation, and result fusion) to treat the best kind of visual feature vectors and linguistic features. We used logistic regression for the predictions. We performed a 10-fold cross validation on the dataset, and we used the average performance to evaluate our model.

Keeping in view the importance of textual information in identifying metaphors, we considered several baseline methods such as DPCNN, Transformer-Endoder(TE), Capsule Network(CN) and TRAT-LSTM(TL). DPCNN is a widely used neural network for text classification designed by Johnson and Zhang [44], Transformer-Encoder is the encoder of a Transformer structure designed by Vaswani et al. [45]. Capsule Network has been proved to be effective in text classification tasks [46]. TRAT-LSTM is a metaphor recognition model designed by Peng et al. [21]. We reconstructed the baselines according to the original author's way.

We quantified and compared our model's performance with the baseline methods on four dimensions: accuracy ($Acc$), $F1$-score ($F1$), precision ($P$), and recall ($R$) to obtain the sentiment classification result. We also compared the performance of fusion model with unimodal models using linguistic or visual features individually.

Table II shows the prediction results using visual features only. Amongst the performance of the five types of visual features, classification accuracy is highest on the second cluster center in the three methods. This indicates that the cluster centers of facial photos of a sentence can capture expressions better. However, the performance using the first and third cluster center is not satisfactory, which may support our assumption that the expressions in the pre-change and post-change phases are not obvious for many participants. It also shows that images of participants' expressions are not always in the middle of sequences. As the time set for taking pictures was at 1-2 seconds after the participants finished reading, the participants' expressions change remained uncontrollable. It is difficult to catch such moments by using images in the middle of sequences.

TABLE II
THE PREDICTION PERFORMANCE OF VISUAL FEATURES. CC 1, CC 2 CC 3 INDICATE THE FIRST, SECOND, AND THIRD CLUSTER CENTER.

| Method | $Acc$ | $F1$ | $P$ | $R$ |
|---|---|---|---|---|
| CC 1 | 0.54 | 0.38 | 0.35 | 0.41 |
| CC 2 | 0.56 | 0.46 | 0.48 | 0.46 |
| CC 3 | 0.48 | 0.22 | 0.16 | 0.33 |
| Mid 3 | 0.55 | 0.46 | 0.46 | 0.47 |
| Mid 20% | 0.55 | 0.46 | 0.46 | 0.47 |

Only using visual features cannot classify sentiments of corresponding sentence precisely, which may be due to the fact that the expression changes of participants are not obvious. However, it is worth noting that using CC 2 visual features outperforms other methods. Thus in the following feature fusion steps, we use the CC 2 visual features as the represent of images of participants' expressions and fuse it with linguistic features.

TABLE III
COMPARISON OF EXPERIMENT RESULTS. FFA, FFC, RF ARE THREE MODELS USE BOTH VISUAL FEATURES AND TEXT FEATURES.

| Method | $Acc$ | $F1$ | $P$ | $R$ |
|---|---|---|---|---|
| FFA | 0.72 | 0.73 | 0.74 | 0.72 |
| FFC | 0.72 | 0.73 | 0.72 | 0.72 |
| RF | **0.77** | **0.77** | **0.78** | **0.75** |
| Visual | 0.56 | 0.46 | 0.48 | 0.46 |
| Text | 0.71 | 0.71 | 0.71 | 0.71 |
| DPCNN | 0.71 | 0.61 | 0.62 | 0.61 |
| TE | 0.75 | 0.62 | 0.63 | 0.61 |
| CN | 0.75 | 0.69 | 0.68 | 0.70 |
| TL | 0.74 | 0.61 | 0.63 | 0.61 |

Table III shows the prediction results of all multimodal and unimodal models. Visual features used here is CC 2. The prediction performance of multimodal method FFA and FFC were not that satisfactory when evaluated on prediction accuracy, while RF significantly outperformed the baseline and unimodal methods in all indicators. We notice that if only textual features are considered, this can improve the $F1$-score significantly but if both visual and textual features are combined together, the prediction performance can be improved enormously.

The overall performance of FFC is better than FFA. However, the highest values of these four indicators are same, which indicates that extending the number of features alone

cannot improve the prediction accuracy. Besides, the time cost of training a model with more features is higher. When the prediction accuracy is similar, the faster method is preferred. In our experiments, FFA was better than FFC.

The prediction accuracies of FFA and FFC were lower than 0.74, but they outperformed the baseline method in recall and precision, so the baseline method suffers overfitting on our dataset and we can diminish the overfitting significantly by adding features and using a simpler model. The prediction accuracy of RF is higher than the baseline method by up to three percentage points. In addition, $P$, $R$, and $F1$ of the RF method are also up by more than 10 percentage points.

Of the three feature fusion methods, RF performs the best. This indicates that classifying sentiment in metaphors using the same model with different methods can improve the results and correct some mistakes. Summing the probabilities of the kinds of sentiments in the target sentence using different features and weights can improve the prediction performance significantly, especially $F1$ and the corresponding $P$ and $R$.

TABLE IV
WEIGHTS OF PREDICTION RESULTS USING DIFFERENT VISUAL FEATURES. CC 1, CC 2 CC 3 INDICATE THE FIRST, SECOND, AND THIRD CLUSTER CENTER.

| | Text | Visual |
|---|---|---|
| CC 1 | 0.5 | 0.5 |
| CC 2 | 0.5 | 0.5 |
| CC 3 | 0.6 | 0.4 |
| Mid 3 | 0.5 | 0.5 |
| Mid 20% | 0.7 | 0.3 |

In the RF method, the weights of prediction results of using visual and linguistic features indicated the importance of the corresponding features. Here we tested fusing prediction result with all kinds of visual features and linguistic features. The weights of different features are in Table IV. These weights are equal when using the visual features of CC 1, CC 2, and Mid 3. Moreover, the highest precision comes on fusing prediction results of linguistic and CC 1 visual features ($P = 0.81$), while fusing with the prediction results of CC 2 visual features obtains the best performance in the other three indicators (Table III). These results indicate that linguistic and visual features are equally important when classifying the sentiments of metaphor.

In all the experiments, we tried to improve the performance of classifying sentiments of metaphors by both heightening the accuracy of prediction and avoiding overfitting. The results show that adding additional information on facial expressions can significantly improve the model's generalizability. Fusing the prediction results of the same model trained by different features can also improve the accuracy of the results while avoiding overfitting. These results support that our method of fusing linguistic and visual features is effective in classifying sentiments of metaphors.

## VI. CONCLUSION

We used facial expression and linguistic features to analyse sentiments of metaphor. We employed a feature fusion approach to fuse linguistic and visual features to improve the performance of classifying sentiments of metaphor. We believe we are the first to use facial expression features to analyze sentiments in metaphors, which supports that when participants' expression changes when reading the metaphor, this information can be useful in detecting sentiments of metaphor. Our study opens new doors in this research field.

Our experimental results show that this model outperforms linguistic and visual models separately. Our approach significantly outperforms the state-of-the-art method. Collecting images of human facial expressions after reading text is inspiring but painful. Due to the rarity of relevant work, such a creation of dataset that includes sentiments of metaphor and the corresponding facial expressions of participants after reading them, is very worth mentioning and innovative.

This paper shows the importance of facial expressions in sentiment analysis of text, which often focuses on linguistic features. Our study has revealed that through the application of appropriate fusion methods, visual features can greatly improve the classification of metaphor sentiments. We hope that this approach inspires those working on text or image classification in future.

## ACKNOWLEDGMENT

This work is partially supported by National Natural Science Foundation of China under Grants No. 62076051. We would like to thank Jie Hou for help with the first draft.

## REFERENCES

[1] T. S. Champlin, "Metaphors we live by." *Philosophical Books*, vol. 23, no. 2, pp. 111–116, 1982.

[2] D. Zhang, M. Zhang, C. Peng, J. J. Jung, and F. Xia, "Metaphor research in the 21st century: A bibliographic analysis," *Computer Science and Information Systems*, 2021.

[3] C. Peng, D. T. Vu, and J. J. Jung, "Knowledge graph-based metaphor representation for literature understanding," *Digital Scholarship in the Humanities*, vol. 18, no. 1, pp. 303–321, 2021.

[4] M. Andrew, E. M. Peter, R. Gnanathusharan, and R. Judy, "Identifying embodied metaphors for computing education," *Computers in Human Behavior*, vol. 105, p. 105859, 2020.

[5] M. Fuyama, H. Saigo, and T. Takahashi, "A category theoretic approach to metaphor comprehension: Theory of indeterminate natural transformation," *Biosyst.*, vol. 197, p. 104213, 2020.

[6] M. M. Hassan, M. G. R. Alam, M. Z. Uddin, S. Huda, A. Almogren, and G. Fortino, "Human emotion recognition using deep belief network architecture," *Information Fusion*, vol. 51, pp. 10–18, 2019.

[7] C. Sharma, D. Bhageria, W. Scott, P. Srinivas, A. Das, T. Chakraborty, V. Pulabaigari, and B. Gambäck, "Semeval-2020 task 8: Memotion analysis - the visuo-lingual metaphor!" *CoRR*, vol. abs/2008.03781, 2020. [Online]. Available: https://arxiv.org/abs/2008.03781

[8] F. M. M. Citron and A. E. Goldberg, "Metaphorical sentences are more emotionally engaging than their literal counterparts," *Journal of Cognitive Neuroscience*, vol. 26, no. 11, p. 2585, 2014.

[9] J. Liu, X. Kong, F. Xia, X. Bai, L. Wang, Q. Qing, and I. Lee, "Artificial intelligence in the 21st century," *IEEE Access*, vol. 6, pp. 34 403–34 421, 2018.

[10] L. Yue, W. Chen, X. Li, W. Zuo, and M. Yin, "A survey of sentiment analysis in social media," *Knowledge and Information Systems*, pp. 1–47, 2018.

[11] S. Steven, D. Kresimir, L. Edward W, L. Robert, and C. Nicola S, "Is language required to represent others' mental states? evidence from beliefs and other representations," *Cognitive science*, vol. 43, no. 1, pp. 2950–2975, 2019.

[12] B. S. and K. M., "Novel ogbee-based feature selection and feature-level fusion with MLP neural network for social media multimodal sentiment analysis," *Soft Comput.*, vol. 24, no. 24, pp. 18 431–18 445, 2020.

[13] J. Xu, Z. Li, F. Huang, C. Li, and P. S. Yu, "Social image sentiment analysis by exploiting multimodal content and heterogeneous relations," *IEEE Trans. Ind. Informatics*, vol. 17, no. 4, pp. 2974–2982, 2021.

[14] K. Zhang, Y. Geng, J. Zhao, J. Liu, and W. Li, "Sentiment analysis of social media via multimodal feature fusion," *Symmetry*, vol. 12, no. 12, p. 2010, 2020.

[15] Q. Li, D. Gkoumas, C. Lioma, and M. Melucci, "Quantum-inspired multimodal fusion for video sentiment analysis," *Inf. Fusion*, vol. 65, pp. 58–71, 2021.

[16] M. Cao, Y. Zhu, W. Gao, M. Li, and S. Wang, "Various syncretic co-attention network for multimodal sentiment analysis," *Concurr. Comput. Pract. Exp.*, vol. 32, no. 24, 2020.

[17] T. Jiang, J. Wang, Z. Liu, and Y. Ling, "Fusion-extraction network for multimodal sentiment analysis," in *Advances in Knowledge Discovery and Data Mining - 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11-14, 2020, Proceedings, Part II*, ser. Lecture Notes in Computer Science, H. W. Lauw, R. C. Wong, A. Ntoulas, E. Lim, S. Ng, and S. J. Pan, Eds., vol. 12085. Springer, 2020, pp. 785–797.

[18] A. Ghosh, G. Li, T. Veale, P. Rosso, E. Shutova, J. A. Barnden, and A. Reyes, "Semeval-2015 task 11: Sentiment analysis of figurative language in twitter," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 470–478.

[19] Z. Kozareva, "Multilingual affect polarity and valence prediction in metaphor-rich texts," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 2013, pp. 682–691.

[20] M. Mohler, D. Bracewell, M. Tomlinson, and D. Hinote, "Semantic signatures for example-based linguistic metaphor detection," *Proceedings of the First Workshop on Metaphor in NLP*, pp. 27–35, 2013.

[21] Y. Peng, C. Su, and Y. Chen, "Chinese metaphor sentiment analysis based on attention-based lstm," in *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, 2018, pp. 478–483.

[22] Y. Lu, K. Sakamoto, H. Shibuki, and T. Mori, "Construction of a multilingual annotated corpus for deeper sentiment understanding in social media," *Journal of Natural Language Processing*, vol. 24, no. 2, pp. 205–265, 2017.

[23] S. M. Mohammad, E. Shutova, and P. D. Turney, "Metaphor as a medium for emotion: An empirical study," in *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, 2016, pp. 23–33.

[24] T. Strzalkowski, S. Shaikh, K. Cho, G. A. Broadwell, L. Feldman, S. Taylor, B. Yamrom, T. Liu, I. Cases, Y. Peshkova, and K. Elliot, "Computing affect in metaphors," in *Proceedings of the Second Workshop on Metaphor in NLP*, 2014, pp. 42–51.

[25] H. L. Nguyen, T. D. Nguyen, D. Hwang, and J. J. Jung, "Kelabteam: A statistical approach on figurative language sentiment analysis in twitter," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 679–683.

[26] F. Xia, J. Liu, H. Nie, Y. Fu, L. Wan, and X. Kong, "Random walks: A review of algorithms and applications," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 2, pp. 95–107, 2019.

[27] J. Ren, F. Xia, X. Chen, J. Liu, M. Hou, A. Shehzad, N. Sultanova, and X. Kong, "Matching algorithms: Fundamentals, applications and challenges," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2021.

[28] S. McGillion, H. M. Alonso, and B. Plank, "Cph: Sentiment analysis of figurative language on twitter easypeasy not," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 699–703.

[29] B. G. Patra, S. Mazumdar, D. Das, P. Rosso, and S. Bandyopadhyay, "A multilevel approach to sentiment analysis of figurative language in twitter," in *International Conference on Intelligent Text Processing and Computational Linguistics*, 2016, pp. 281–291.

[30] V. Rentoumi, G. Giannakopoulos, V. Karkaletsis, and G. A. Vouros, "Sentiment analysis of figurative language using a word sense disambiguation approach," in *Proceedings of the International Conference RANLP-2009*, 2009, pp. 370–375.

[31] D. Zhang, H. Lin, P. Zheng, L. Yang, and S. Zhang, "The identification of the emotionality of metaphorical expressions based on a manually annotated chinese corpus," *IEEE Access*, vol. 6, pp. 71 241–71 248, 2018.

[32] V. Dankers, M. Rei, M. Lewis, and E. Shutova, "Modelling the interplay of metaphor and emotion through multitask learning," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, Nov. 2019, pp. 2218–2229.

[33] P.-L. Huguet Cabot, V. Dankers, D. Abadi, A. Fischer, and E. Shutova, "The Pragmatics behind Politics: Modelling Metaphor, Framing and Emotion in Political Discourse," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online, Nov. 2020, pp. 4479–4488.

[34] "Facial expression recognition using iterative universum twin support vector machine," *Applied Soft Computing*, vol. 76, pp. 53–67, 2019.

[35] D. Gupta, B. Richhariya, and P. Borah, "A fuzzy twin support vector machine based on information entropy for class imbalance learning," *Neural Computing and Applications*, vol. 31, no. 11, pp. 7153–7164, 2019.

[36] D. Gupta and B. Richhariya, "Entropy based fuzzy least squares twin support vector machine for class imbalance learning," *Applied Intelligence*, vol. 48, no. 11, pp. 4212–4231, 2018.

[37] D. Zhang, H. Lin, L. Yang, S. Zhang, and B. Xu, "Construction of a chinese corpus for the analysis of the emotionality of metaphorical expressions," in *ACL 2018: 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 144–150.

[38] S. Zhang and P. Huang, "High-resolution, real-time 3d shape acquisition," in *2004 Conference on Computer Vision and Pattern Recognition Workshop*, 2004, pp. 28–28.

[39] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, no. 3, pp. 1755–1758, 2009.

[40] J. Liu, F. Xia, L. Wang, B. Xu, X. Kong, H. Tong, and I. King, "Shifu2: A network representation learning based model for advisor-advisee relationship mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 4, pp. 1763–1777, 2021.

[41] M. Hou, J. Ren, D. Zhang, X. Kong, D. Zhang, and F. Xia, "Network embedding: Taxonomies, frameworks and applications," *Computer Science Review*, vol. 38, p. 100296, 2020.

[42] K. Sun, L. Wang, B. Xu, W. Zhao, S. W. Teng, and F. Xia, "Network representation learning: From traditional feature learning to deep learning," *IEEE Access*, vol. 8, no. 1, pp. 205 600–205 617, 2020.

[43] E. Shutova, D. Kiela, and J. Maillard, "Black holes and white rabbits: Metaphor identification with visual features." in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 160–170.

[44] R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017, pp. 562–570.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[46] W. Zhao, J. Ye, M. Yang, Z. Lei, S. Zhang, and Z. Zhao, "Investigating capsule networks with dynamic routing for text classification," *arXiv preprint arXiv:1804.00538*, 2018.