EKCP:Emoji Knowledge Contained Pre-training for Sentiment Analysis

Anonymous EMNLP submission

Abstract

Recently, the proportion of emojis that has strong correlation with sentiment orientation is gradually increasing in the text. However, since the lexicon of pre-training models like BERT regards almost all emojis as [UNK], the sentiment knowledge contained in emojis is ignored during pre-training. Furthermore, even if emojis are included in the lexicon to treat emoji as a normal word, the pre-training model cannot fully mine the sentiment information contained in emoji, because emojis have a greater impact on the sentiment of the text than normal word. Therefore, focusing the Chinese social media platform, Emoji Knowledge Contained Pre-training (EKCP) is proposed to take emojis into consideration. Specifically, we built two lexicons to conduct emoji-focused masking and emoji combination masking, and then construct two prediction objectives based on the two masking, aiming to embed sentiment information of not only a single emoji but also the emoji combination into pre-trained sentiment representation. In addition, a realistic and challenging Chinese sentiment analysis dataset including emoji in text is created for the first time by hand-tagging 50k samples from social platforms, forums and e-commerce platforms. Finally, a series of experiments on the developed dataset show that EKCP significantly outperforms strong pre-training baseline.

1 Introduction

Sentiment analysis that has a wide range of applications in data mining (Liu, 2011; Zhang et al., 2018), information retrieval, refers to that classify the sentiment or opinion of text into two or more types.

In recent years, sentiment analysis has made great progress, which is mainly due to two aspects. On the one hand, with the rapid development of social media on the Internet, such as product reviews and forum discussions, there are more text data available for model training. On the other hand, the extensive using of pre-training models (Peters et al., 2018; Vaswani et al., 2017) such as BERT (Devlin et al., 2019) and its refined versions (Yang et al., 2019) in recent years has greatly improved the performance of sentiment analysis models.

However, the development of social media (Facebook, Twitter, Weibo, etc.) not only expands training datasets, but also brings diversification of text forms. More and more texts contain emoji that has a high correlation with sentiment orientation. As shown in the first two rows of Table 1, with or without emoji, the same sentence expresses completely different sentiment. Nevertheless, since the lexicon of the previous pre-trained models do not contain emoji, they can only convert emoji to [UNK], which completely ignores the information contained in emoji.

Thus, we propose that the emoji should be taken into account when pre-training. Before us, only few works have considered emoji (Barbieri et al., 2016; Eisner et al., 2016; Chen et al., 2019; Shoeb et al., 2019; Kralj Novak et al., 2015; Ge, 2019). But although these methods prove the importance of emoji for sentiment analysis, they did not consider how to embed the sentiment information in emoji into the sentence representation of the pre-training model. Obviously, the most simple way is to include emojis in the lexicon of the pre-training models so that the model can treat emoji as a normal word. But treating emoji and other words equally can not fully mine the sentiment information contained in emoji because emoji has a stronger influence on the sentiment of text than other normal word. Moreover, these previous methods regard emoji as a single word, which ignores the information contained in the emoji combination. Similar to the combination of different words to form a phrase will derive another meaning, the combination of emoji will also represent different semantics. The last two rows of Tabel 1 illustrate two different

Table 1: Comparison of the way about EKCP and BERT to process emoji. "👍" and "💩" represent good and bad respectively in Chinese usage habits. Without the two emojis, the text in first and second lines will become a neutral statement. The text in the third row is a typical product review on a Chinese e-commerce platform, different numbers of "⭐" represents different sentiment. More than three "⭐" represents positive reviews, less than three "⭐" is the opposite. The combination of "🐮" and "🍺" evolves from spoken language, representing the thing or person is very powerful.

Text	Model view	Prediction
手机信号:👍 (Phone signal: 👍)	**BERT:**手机信号: [UNK] **EKCP:**手机信号: 👍	neutral positive
给个表情💩 自己体会 (Give you a symbol 💩, and you will know it)	**BERT:**给个表情[UNK]自己体会 **EKCP:**给个表情💩 自己体会	neutral negative
手机拍照效果:⭐⭐⭐⭐⭐ (The performance of the photos taken by the phone: ⭐⭐⭐⭐⭐)	**BERT:**手机拍照效果: [UNK] **EKCP:**手机拍照效果: ⭐⭐⭐⭐⭐	neutral positive
使用起来感觉非常:🐮🍺 (It is very 🐮🍺 to use)	**BERT:**使用起来感觉非常: [UNK] **EKCP:**使用起来感觉非常: 🐮🍺	neutral positive

emoji combinations.

Therefore, we propose EKCP to learn a Chinese sentence representation containing emoji and emoji combination information. Specifically, there are two lexicons in EKCP: 1) a lexicon that consists of normal words and frequently-used emojis and 2) emoji combination lexicon derived from unsupervised method (Section 3.1). With the help of the two lexicons, we conceal the information of single emoji and emoji combination though emoji-focused masking and emoji combination masking (Section 3.2). Subsequently, the pre-training model is trained to recover these information with two prediction objectives (Section 3.3). Compared with the traditional pre-training model, EKCP not only integrates the information of normal vocabulary but also focuses on embedding the information in emoji and emoji combinations that have a strong correlation with sentiment orientation into the pre-trained sentence representation.

To evaluate the proposed method over a variety of different scenarios, a Chinese sentiment analysis dataset including emoji is created for the first time. To the best of our knowledge, available datasets including emojis are collected from single platform like Twitter (Kralj Novak et al., 2015; Eisner et al., 2016) and not in Chinese. Different from existing released datasets, the sentiment analysis dataset we propose is collected from major Chinese e-commerce platforms and forums like Taobao, Weibo, Zhihu and so on, which not only contains more and richer scenes, but also fills in the gaps that there is no Chinese sentiment analysis dataset including emojis.

In general, our contributions can be summarized as follows:

1) The Emoji Knowledge Contained Pre-training for sentiment analysis is proposed to embed the information contained in single emoji and emoji combination into the representation of the Chinese sentence, which is the first research on a pre-training model considering emoji associated with BERT as far as we know.

2) We develop a Chinese sentiment analysis dataset that includes the text containing emojis for the first time, aiming to test the effectiveness of our model. The dataset is available at `https://github.com/hellonlp/ekcp/blob/main/data/sa.xlsx`

3) EKCP achieves state-of-the-art result on our sentiment dataset compared with the previous excellent pre-trained model (Eisner et al., 2016; Illendula and Yedulla, 2018).

2 Related works

2.1 Sentiment Analysis

A large number of works have been developed for sentiment analysis, including supervised and unsupervised methods. In supervised methods, traditional machine learning methods such as Support

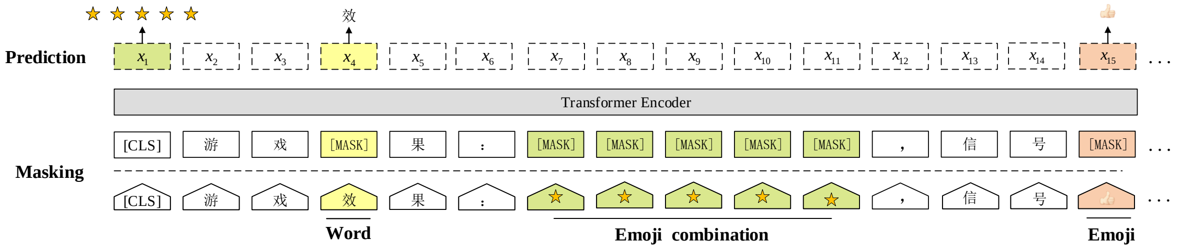
效
Prediction
x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10} x_{11} x_{12} x_{13} x_{14} x_{15} . . .
Transformer Encoder
[CLS] 游 戏 [MASK] 果 : [MASK] [MASK] [MASK] [MASK] [MASK] , 信 号 [MASK] . . .
Masking
[CLS] 游 戏 效 果 : , 信 号 . . .
Word
Emoji combination
Emoji

Figure 1: The structure of Emoji Knowledge Contained Pre-training (EKCP). There are two stages in EKCP: (1) **Masking**. EKCP produces a corrupted version of an input sequence by removing the information of masked tokens. (2) **Prediction**. Emoji Combination (EC) prediction (on x_1) and Emoji-Focused (EF) prediction (on x_4 and x_{15}) are jointly optimized so as to recover the removed information from the corrupted version. Note that, since the masked emoji tokens have been predicted in the emoji combination prediction (on x_1), EF is not need to calculate on x_7,x_8,x_9,x_{10} and x_{11}.

Vector Machine (Moraes et al., 2013; Ye et al., 2005) and Naive Bayes (Narayanan et al., 2013) are used in sentiment analysis. Unsupervised methods mostly use the sentiment lexicon, semantic dependency parsing and so on to make sentiment judgment. (Wilson et al., 2005; Wang et al., 2016)

With the development of deep learning, the use of recurrent neural network (Tai et al., 2015; Xu et al., 2016) and Convolutional Neural Networks (Cai and Xia, 2015; Liao et al., 2017; Bin et al., 2017) for sentiment analysis has achieved great success. Subsequently, the proposal of BERT and its optimized version has once again improved the accuracy of sentiment analysis by leaps and bounds (Jiang et al., 2019; Raffel et al., 2019; Xie et al., 2019; Sun et al., 2020).

2.2 Sentiment in Emoji

Emojis are widely used in daily life, and the information hidden in emoji can be used for sentiment analysis. The first study in this filed is conducted by Novak (Kralj Novak et al., 2015). In addition, there is also an informal blog post published by the Instagram Data Team to study emoji (Dimson, 2015). Their research provides valuable insights for the use of emoji on Instagram, and shows that emoji representation helps to understand the sentiment of the sentence. Barbieri (Barbieri et al., 2016) used the skip-gram (Mikolov et al., 2013) to train emoji embedding from a large Twitter datasets. In terms of sentiment analysis, Eisner (Eisner et al., 2016) also utilized Twitter data as training set, and conducts experiments sentiment classification task, which proves the usefulness of emoji representation. And Chen (Chen et al., 2019) proposes a novel representation learning method that uses emoji prediction as an instrument to learn respective sentiment-aware representations for each language. The learned representations are then integrated to facilitate cross-lingual sentiment classification. Moreover, the information available in emoji not only exists in a single emoji, but also among multiple emojis. These information can be learned through the co-occurrence network (Illendula and Yedulla, 2018) which can embed emojis into a low dimensional vector space so as to address sentiment task including emojis.

2.3 Pre-training Approaches

Pre-training language model (LM) refers to obtaining language representations from large-scale data through self-supervised learning. Before BERT, almost all language models are Auto-regressive LM which predicts the next token based on the above or predicts the previous token based on the following (Bengio et al., 2003; Pennington et al., 2014; Peters et al., 2018; Radford et al., 2018). Autoencoder LMs represented by BERT construct a self-surpervised objective called masked language model to pre-train the encoder, and relies only on large-size unlabeled data (Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019; Lan et al., 2019). Among them, BERT uses a bidirectional Transformer (Vaswani et al., 2017) as a feature extractor, which can not only obtain longer contextual information, but also improve the feature extraction capability compared with traditional neural network. After that, LM randomly masks some tokens (default mask 15% tokens) and taking the whole sequence together as input, then calculates the loss of masked tokens during pre-training. On the shoulders of BERT, RoBERTa (Liu et al., 2019) adjusts hyper-parameters and expands the dataset to gain performance beyond BERT.

3 EKCP: Emoji Knowledge Contained Pre-training

In many cases, emoji and its combinations have a great influence on text sentiment. In order to make use of these influence, EKCP is proposed to make additional enhancements for pre-training model by involving the information contained in emoji and emoji combination during pre-training.

Specifically, first of all, commonly used emojis are added to the basic lexicon of BERT so that the pre-trained model can treat different emojis as different tokens. And then, we need to obtain a lexicon of meaningful emoji combinations through an unsupervised method (Section 3.1). Subsequently, the two kind of masking (Section 3.2) are introduced to remove information contained in emoji and emoji combination with the help of basic lexicon containing emoji and the emoji combination lexicon. Finally, two pre-training objectives (Section 3.3) are proposed to enforce pre-trained model to recover the masked information from the corrupted version.

Formally, the masking constructs a corrupted version $\tilde{X}$ for an input sequence X. x_i and $\tilde{x}_i$ denote the i-th token of X and $\tilde{X}$ respectively. After masking, a parallel data ($\tilde{X}$,X) is obtained. Thus, the transformer encoder can be trained with pre-training objectives that are supervised by recovering masked information using the final states of encoder $\tilde{\text{x}}_1$,...,$\tilde{\text{x}}_n$.

3.1 Unsupervised Emoji Combination Knowledge Mining

As shown in the last two rows of Table 1, the combination of emojis produces completely different sentiment orientations. In order to mine the information of emoji combinations, self-information (Katona and Nemetz, 1976; Luo and Sun, 2003) and mutual information (Finn, 1993) that do not require any supervision are adopted in EKCP. Self-information reflects the richness of tokens adjacent to the current emoji combination, while mutual information is a manifestation of the dependence between different emojis within the emoji combination. In the end, we comprehensively consider the self-information and mutual information to obtain the score that multiple emojis can become an emoji combination.

Self-information Using self-information (Katona and Nemetz, 1976; Luo and Sun, 2003), we can get the richness of left neighbor tokens set and right neighbor tokens set. In particular, define $C = (t_j, t_{j+1}, ..., t_{j+a-1})$ as possible emoji combinations in the input text, where j and a indicate the starting index in the text and the length of the emoji combination, respectively. Note that the starting index of the same C in different texts is different. We count the left neighbor tokens set $W^{left} = \{t^1_{j-1}, t^2_{j-1}, ..., t^m_{j-1}\}$ and right neighbor tokens set $W^{right} = \{t^1_{j+a}, t^2_{j+a}, ..., t^k_{j+a}\}$ of C in all corpora, where m and k refer to total number of left neighbor tokens and right neighbor tokens respectively. The formula for calculating the richness of W^{left} can be expressed as:

$$EL = -\sum_{w_j \in W^{left}} P(w_j|C) log P(w_j|C) \quad (1)$$

The richness ER of W^{left} is consistent with Equation (1). The higher the EL and ER, the higher the uncertainty of the tokens that appears on the left and right sides of C, which means the probability that C independently becomes a meaningful emoji combination becomes greater.

Once EL and ER are obtained, we can apply

$$\begin{aligned} R(C) = ER \cdot log(\frac{EL}{|EL - ER|}) \\ + EL \cdot log(\frac{ER}{|EL - ER|}) \end{aligned} \quad (2)$$

to comprehensively consider the richness of the left neighbor tokens set and right neighbor tokens. Obviously, R will be larger only when both EL and ER are larger.

Mutual information Average Mutual Information (AMI) (Finn, 1993) can evaluate the dependencies among different tokens. The higher the AMI, the greater the probability that different tokens can form an emoji combination. AMI can be expressed as:

$$AMI(C) = \frac{p(C)}{a} log(\frac{p(C)}{\prod_{h=j}^{j+a-1} p(t_h)}) \quad (3)$$

Score of Emoji Combination After obtaining the two evaluation indicators, the score of the candidate

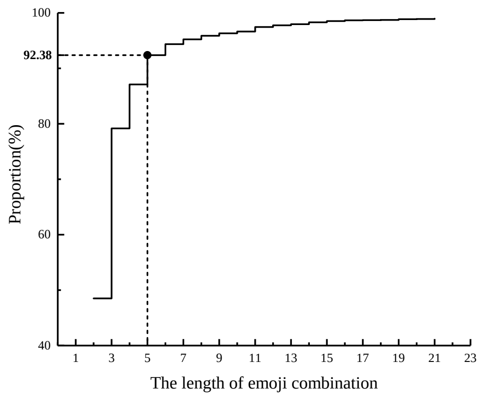

Figure 2: The proportion of emoji combinations that are no more than l to all emoji combinations

emoji combination can be figured out:

$$S = R(C) + AMI(C) \tag{4}$$

And then, according to S, we choose top 2000 emoji combinations as a emoji combination lexicon D.

Hyperparameter Suppose l represents the longest length of the emoji combination. Obviously, $a \leq l$. As shown in Figure 2, when l is set to be greater than 5, the proportion will not be greatly improved. Thus, l is set to 5. In this case, 92.38% of the combinations are included to calculate their score S. Some emoji combination examples will be listed in Appendix.

3.2 Masking

There are two types of masking. Concretely, we conduct emoji combination masking and emoji-focused masking successively like Figure 1.

Emoji Combination Masking The purpose of this step is to mask the information of emoji combination. Based on emoji combination lexicon D, we apply the maximum reverse matching algorithm to select out emoji combinations existing in the text and then replace them with [MASK]. Note, up to 3 emoji combinations in a single text will be randomly selected out to mask.

Emoji-focused Masking In the Emoji-focused Masking stage, we need to do two types masking. If the proportion of masked emoji combinations is less than 15%, we will start to mask a single emoji. After that, in case that the proportion of the masked is still less than 15%, we will mask the non emoji tokens. Finally, all the masked tokens account for no more than 15%.

3.3 Pre-training Objectives

The masking constructs a corrupted version $\tilde{X}$, where their information is substituted with masked tokens. The two objectives are defined to tell the transformer encoder to recover the replaced information. Our objective function consists of two parts, one is emoji-focused prediction L_{ef} and the other is emoji combination prediction L_{ec}. Finally, the objective function we trained is: $L = L_{ef} + L_{ec}$.

Emoji-focused Prediction By using the output vector $\tilde{\mathrm{x}}_i$ from transformer encoder, emoji-focused prediction allows model to infer the masked tokens including emoji and no emoji. Specifically, The vector $\tilde{\mathrm{x}}_i$ is fed into an output softmax layer, which constructs a normalized probability vector $\hat{y}_i$ over the language model lexicon. In such way, the emoji-focused prediction objective L_{ef} is to maximize the probability of original token x_i as follows:

$$\hat{\mathrm{y}}_i = softmax(\tilde{\mathrm{x}}_i \mathrm{W}_{ef} + \mathrm{b}_{ef}) \tag{5}$$

$$L_{ef} = -\sum_{i=1}^{n} \mathrm{y}_i log \hat{\mathrm{y}}_i \tag{6}$$

Here, W_{ef} and b_{ef} are the parameters of the output layer. y_i is the one-hot representation of the original token x_i.

Emoji Combination Prediction Compared with a single emoji, there are more exact sentiment information in emoji combinations. Therefore, in order to capture the dependency among emojis, an emoji combination objective is proposed. Unlike BERT that assumes tokens can be independently predicted, we conduct emoji combination prediction with multi-label classification. Specifically, we input the final state of classification token [CLS] that denotes representation of the entire sequence into the output sigmoid layer to predict multiple labels (emoji combinations in $\tilde{X}$). The emoji combination objective L_{ec} is denoted as follows:

$$\hat{\mathrm{y}}_e = sigmoid(\tilde{\mathrm{x}}_1 \mathrm{W}_{ec} + \mathrm{b}_{ec}) \tag{7}$$

$$L_{ec} = -\sum_{e=1}^{E} \mathrm{y}_e log \hat{\mathrm{y}}_e \tag{8}$$

Here, $\tilde{\mathrm{x}}_1$ denotes the output vector of [CLS]. E is the number of masked emoji combinations in $\tilde{X}$. y_e is the sparse presentation of a target emoji combination. $\hat{\mathrm{y}}_e$ is the token probability after sigmoid. Each element of y_e corresponds to one token of the lexicon D, and is equal to 1 if y_e contains the corresponding token.

4 Experiments

4.1 Dataset and Evaluation metric

Dataset: To perform pre-training, we collect 101,000k Chinese text samples from Chinese social media platform, of which 100,000k samples do not include emoji and the remaining 1,000k samples contain it. As for sentiment analysis task, we gather another 50k Chinese text samples that includes both emoji and general samples and label them with the their corresponding sentiment (positive, neutral, or negative) carefully. A sample will be labeled by three people on average to ensure that the result is objective and accurate. And then, we divide them into training set (40k samples) and test set (10k samples). We display a few specific samples of pre-training dataset and sentiment dataset in Table 2. It can be observed from the samples in the sentiment dataset that both emoji and emoji combination have a strong correlation with the sentiment contained in the text.

Evaluation metric: We report the model performance through accuracy.

4.2 Experiment Setting

All experiments are performed on four NVIDIA GEFORCE GTX 1080Ti GPU and the model is implemented based on Tensorflow (Abadi et al., 2016). The experimental details of pre-training and fine-tuning are described as follows:

Pre-training: RoBERTa$_{base}$, the base version of RoBERTa, is applied as the transformer encoder of our pre-trainning model. Since the limit of GPU resource, we do not try other larger versions of RoBERTa. Except that the batch size is 128 and the model is trained for 600k steps, we keep the rest of setting consistent with RoBERTa$_{base}$.

Fine-tuning: Our model is fine-tuned on the sentence-level sentiment classification task that is to classify the sentiment orientation (positive, neutral, or negative) of an input sentence. When fine-tuning, we feed a sentence into the pre-trained model and apply the final hidden vector $\tilde{\mathrm{x}}_1$ as the aggregate representation of the input. Then, $\tilde{\mathrm{x}}_1$ is fed into a fully connected layer to output the final classification vector $F \in \mathbb{R}^K$, where K is the number of classes. After the model output F, we calculate the standard classification loss function, i.e.,$log(softmax(F))$.

4.3 Ablation experiment

In order to verify the effectiveness of the proposed module, we conduct a large number of ablation experiments and report separately the accuracy of sentiment prediction for text with and without emoji in the test set in Table 3. We apply RoBERTa as our strong baseline. It can be seen that the accuracy is improved greatly on the text with emojis after the emoji is included in lexicon of pre-trained model, which proves that emoji does contain the information needed for sentiment analysis. The accuracy of the model on the text with emojis is further increased after we perform emoji-focused masking. It shows that when using emoji-focused masking, the model can pay more attention to the information from emoji, which is benefit to sentiment analysis. Finally, when both emoji combination masking and emoji-focused masking are performed as shown in last row, the model accuracy on text with emojis increased to 89.42. This result shows that only using emoji masking will split emoji combinations that are strongly correlated with sentiment. Therefore, performing a emoji combination masking before emoji-focused masking can further improve the accuracy of the model on the text with emojis since emoji combination masking reveals the inner connection between emoji.

At the same time, we notice that after using emoji-focused masking and emoji combination masking, the accuracy of texts that do not contain emoji has not changed much. Thus, in general, our model can improve the accuracy of text sentiment analysis.

4.4 Attention Visualization

In order to explain EKCP more intuitively, we visualize the attention distribution of the final hidden vector $\tilde{\mathrm{x}}_1$ in Table 4. We can clearly observe which token has a decisive influence on the final sentiment classification by attention distribution.

For these examples, no matter whether RoBERTa predicts correctly or not, it cannot focus its attention on emoji with rich sentiment information. And when emoji completely dominates the sentiment of the entire sentence, RoBERTa's prediction will become unreliable as shown in the

Table 2: Several samples of the pre-training dataset and sentiment dataset. "👍" and "👌" represent affirmative, "😜" means naughty and happy, and "💔" denotes heartbroken and sad.

Dataset	Emoji	Samples	Sentiment annotation
Pre-training	yes	外形外观：👍👍 好喜欢他的颜色。(Appearance：👍👍 I really like its color.)	/
	yes	电池👌拍照👌 😜😜😜 (Battery👌photograph 👌 😜😜😜)	/
	no	性价比太高了吧，颜值也没得说。(It is cost-effective and pretty.)	/
	no	运行速度：运行流畅，毫无卡顿。(Running speed: smooth running. No lag.)	/
Sentiment	yes	售后服务👌，一次愉快的购物体验 (After-sales service is 👌. A pleasant shopping experience)	positive
	yes	看起来还行，用起来实在是💔💔💔 (It looks okay, it's really 💔💔💔 to use.)	negative
	no	刚下单就收到货了，用起来感觉还不错。(I received the goods as soon as I placed the order, and it feels good to use.)	positive
	no	才用没几天，过一段时间再评价吧。(It's only been used for a few days, let's evaluate it later.)	neutral

Table 3: The accuracy (%) achieved by our model under different settings.

Model	Emoji-contained lexicon	Emoji-focused prediction objective		Emoji combination prediction objective	The form of emoji	Text without emoji	Text with emoji	Overall text
		Random masking	Emoji-focused masking	Emoji combination masking				
Baseline		✓			[UNK]	80.72	84.31	81.45
EKCP	✓	✓			emoji	80.70	88.50	82.21
EKCP	✓		✓		emoji	81.36	89.01	82.84
EKCP	✓		✓	✓	emoji	81.39	89.42	82.94

second example. On the contrary, EKCP can pay attention to the emoji token containing sentiment information. More importantly emoji combination can also be focused to fully dig the sentiment information in the permutation and combination of emoji.

4.5 Comparison with state-of-the-art models

We compare the performance of our model with two typical works considering the influence of emoji on sentiment analysis and several state-of-the-art pre-training models to verify the superiority of EKCP encoding the representation of text containing emoji, and the results are reported the experimental in Table 5. EM and PR represent the sentiment analysis method considering emoji and the most advanced pre-training model, respectively. In EM, emoji2vec (Eisner et al., 2016) constructs a binary classification model to determine whether a sequence of words describing the current emoji are appropriate. There are two inputs of this model: 1) emoji embedding that is initialized randomly and 2) the descriptive words embedding that employ public word2vec (Tomas Mikolov and Dean, 2013). Illendula et.el (Illendula and Yedulla, 2018) use text containing emoji to train emoji co-occurrence network to get emoji embeddings. Following their settings, we obtain the corresponding emoji embeddings and then get prediction accuracy on sentiment analysis based on these embeddings. It can be seen that because the two methods are not combined with the pre-training model, the accuracy is not very satisfactory.

In order to make a fair comparison with state-of-the-art pre-training models: 1) We select the base version of each pre-training model in Table 4 so as that the parameter amount of all the model is not much different; 2) All parameters of the model are initialized randomly, and then we pre-train and fine-tune the models on our pre-training dataset and sentiment dataset containing emoji respectively.

It can be observed that the model performance of EKCP is better than that of BERT, RoBERTa and ALBERT. This is because these three pre-training models directly transform the emoji token into [UNK] since the lexicon of them do not contain emoji, which leads to the loss of information carried by emoji with strong sentiment tendency. On the contrary, EKCP first constructs a lexicon containing frequently-used emoji to ensure that the emoji information will not be lost. And then the emoji-focused prediction objective and emoji combination prediction objective can futher explore the sentiment orientation contained in emoji.

5 Conclusion

Through EKCP, we deepen our understanding of emoji knowledge and apply it to pre-trained language models. With the help of the extended lex-

Table 4: Visualization of chosen samples. The darker the color, the greater the attention weight (For ease of understanding, the English translation of the Chinese in the table is given here, the translation of ’这款手机拍照效果’ is "The camera effect of this phone", the translation of ’客服服务态度’ is "The customer service attitude", the translation of ’购买十天后显示器到货’ is "The monitor arrives ten days after purchase", the translation of ’用起来’ is ’The experience of use’.)

No	Model	Sentence Samples	Prediction
1	RoBERTa	这款手机拍照效果😘😘😘，客服服务态度⭐⭐⭐⭐⭐	**neutral**
2	EKCP	这款手机拍照效果😘😘😘，客服服务态度⭐⭐⭐⭐⭐	**positive**
3	RoBERTa	购买十天后显示器到货😭😭😭，用起来👎👎	**positive**
4	EKCP	购买十天后显示器到货😭😭😭，用起来👎👎	**negative**

Table 5: Comparison of EKCP with other pre-trained models

	Model	Acc
EM	emoji2vec	75.08
	Illendula et.el	73.95
PR	BERT*base*	81.36
	ALBERT*base*	81.32
	RoBERTa*base*	81.45
Our model	EKCP	82.94

icon and emoji combination lexicon, as well as the two mask methods, EKCP has achieved a very in-depth understanding of emoji. Therefore, with the widespread use of social media today, EKCP’s understanding of text has been greatly improved. While ensuring the effect of regular text, EKCP has greatly improved accuracy of text data containing emoji in the sentiment analysis task, and surpasses that of RoBERTa. In the future, EKCP can not only be applied to sentiment analysis tasks, but also will be of great help to all NLP tasks that need to understand emoji semantics.

Acknowledgements

We employed 5 persons who are familiar with the sentiment of comments. Each person labels 30K, so a total of 150k data is labeled. Each sentence will be labeled by 3 persons, and the labels shall be determined according to most principles.

Table 6: Some interesting emoji combinations

Emoji combination	Sentiment
⭐⭐	negative
⭐⭐⭐⭐⭐	positive
🐮🍺	positive
🌶🐔	negative
🍐🎼	negative

Appendix

In Table 6, we display some interesting emoji combinations which are found by unsupervised emoji combination knowledge mining. We further found the original text containing emoji combinations to analyze the derivation of these emoji combinations. The generation of different number of "⭐" is because the website requires different numbers of stars to express the degree of satisfaction with the product when the user evaluates the purchased product. The "🐮🍺", "🌶🐔", and "🍐🎼" are all due to users’ pursuit of fast and convenient communication. For example, "🌶🐔" represents "垃圾" (rubbish) in Chinese. Since users find it too troublesome to type these two Chinese characters through pinyin and "🌶🐔" in Chinese semantics reflects the two words "垃圾" vividly, the combination of "🌶🐔" appears in large numbers in the Internet.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283.

Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. 2016. What does this emoji mean? a vector space skip-gram model for twitter emojis. In *LREC 2016*, page 3111–3119.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The journal of machine learning research*, 3:1137–1155.

Liang Bin, Liu Quan, Xu Jin, Zhou Qian, and Zhang Peng. 2017. Aspect-based sentiment analysis based on multi-attention cnn. *Journal of Computer Research and Development*, 54(8):1724.

Guoyong Cai and Binbin Xia. 2015. Convolutional neural networks for multimedia sentiment analysis. In *Natural Language Processing and Chinese Computing*, pages 159–167. Springer.

Zhenpeng Chen, Sheng Shen, Ziniu Hu, Xuan Lu, and Xuanzhe Liu. 2019. Emoji-powered representation learning for cross-lingual sentiment classification. In *WWW' 19*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas Dimson. 2015. Machine learning for emoji trends. *http://instagramengineering.tumblr.com/post/117889701472/emojineering-part-1-machine-learning-for-emoji*.

Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bošnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. *WS*.

John T Finn. 1993. Use of the average mutual information index in evaluating classification error and consistency. *International Journal of Geographical Information Science*, 7(4):349–366.

Jing Ge. 2019. Emoji sequence use in enacting personal identity. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 426–438, New York, NY, USA. Association for Computing Machinery.

Anurag Illendula and Manish Reddy Yedulla. 2018. Learning emoji embeddings using emoji co-occurrence network graph. *arXiv preprint arXiv:1806.07785*.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. 2019. Smart: Robust and efficient fine-tuning for pretrained natural language models through principled regularized optimization. *arXiv preprint arXiv:1911.03437*.

Gyula Katona and O Nemetz. 1976. Huffman codes and self-information. *IEEE Transactions on Information Theory*, 22(3):337–340.

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

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Shiyang Liao, Junbo Wang, Ruiyun Yu, Koichi Sato, and Zixue Cheng. 2017. Cnn for situations understanding based on sentiment analysis of twitter data. *Procedia computer science*, 111:376–381.

Bing Liu. 2011. Sentiment analysis and opinion mining. In *Synthesis Lectures on Human Language Technologies 5.1 (2012): 1-167*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

S. Luo and M. Sun. 2003. Two-character chinese word extraction based on hybrid of internal and contextual measures. *Association for Computational Linguistics*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*.

Rodrigo Moraes, João Francisco Valiati, and Wilson P GaviãO Neto. 2013. Document-level sentiment classification: An empirical comparison between svm and ann. *Expert Systems with Applications*, 40(2):621–633.

Vivek Narayanan, Ishan Arora, and Arjun Bhatia. 2013. Fast and accurate sentiment classification using an enhanced naive bayes model. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 194–201. Springer.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Abu Awal Md Shoeb, Shahab Raji, and Gerard de Melo. 2019. EmoTag – Towards an emotion-based analysis of emojis. In *Proceedings of RANLP 2019*, pages 1094–1103.

Zijun Sun, Chun Fan, Qinghong Han, Xiaofei Sun, Yuxian Meng, Fei Wu, and Jiwei Li. 2020. Self-explaining structures improve nlp models. *arXiv preprint arXiv:2012.01786*.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Kai Chen Greg S Corrado Tomas Mikolov, Ilya Sutskever and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Conference and Workshop on Neural Information Processing Systems*.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. Opinionfinder: A system for subjectivity analysis. In *Proceedings of HLT/EMNLP 2005 Interactive Demonstrations*, pages 34–35.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2019. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*.

Jiacheng Xu, Danlu Chen, Xipeng Qiu, and Xuangjing Huang. 2016. Cached long short-term memory neural networks for document-level sentiment classification. *arXiv preprint arXiv:1610.04989*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Conference and Workshop on Neural Information Processing Systems*.

Qiang Ye, Bin Lin, and Yi-Jun Li. 2005. Sentiment classification for chinese reviews: A comparison between svm and semantic approaches. In *2005 International Conference on Machine Learning and Cybernetics*, volume 4, pages 2341–2346. IEEE.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis : A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, page e1253.