SELF-SUPERVISED LEARNING FOR SENTIMENT ANALYSIS VIA IMAGE-TEXT MATCHING

Haidong Zhu[⋆] *Zhaoheng Zheng*[⋆] *Mohammad Soleymani*[⋆,†] *Ram Nevatia*[⋆]

[⋆]University of Southern California [†]USC-ICT

ABSTRACT

There is often a resemblance in the sentiment expressed in social media posts (text) and their accompanying images. In this paper, We leverage this sentiment congruence for self-supervised representation learning for sentiment analysis. By teaching the model to pair an image with its corresponding social media post, the model can learn a representation capturing sentiment features from the image and text without supervision. We then use the pre-trained encoder for feature extraction for sentiment analysis in downstream tasks. We show significant improvement and good transferability for sentiment classification in addition to robustness in performance when available data decreases on public datasets (B-T4SA and IMDb Movie Review). With this work, we demonstrate the effectiveness of self-supervised learning through cross-modal matching for sentiment analysis.

Index Terms— Sentiment analysis, image-text matching, self-supervised learning

1. INTRODUCTION

The sentiment is an affective disposition that represents our attitude towards an entity. Sentiment analysis is the automatic identification of the polarity of the sentiment from language or other available modalities [1]. Sentiment analysis is widely used for tasks such as opinion mining [2].

In social media posts, the sentiment expressed in text and accompanying multimedia content is often congruent. Thus, when a model is learning to determine whether a text and an image are from the same post, the sentiment information is captured in the representation from this association. Figure 1 illustrates examples of matching and non-matching images. In this example, when deciding which one of the two images describes a wedding ceremony, we can compare their sentiments to make the distinction. The sentiment might not be apparent when we identify the correct pairs; nevertheless, learning the text-image implicit association allows sentiment-relevant feature learning.

We propose using image-text matching as the pretext task for sentiment analysis. We call our method **Senti**ment analysis via **I**mage-**TE**xt **M**atching, in short, Senti-ITEM. Specifically, we train the model to pair the text with the correct image and use this pretrained model to generate features for the image or the text for sentiment classification. During the training of the image-text matching task, no sentiment information is available as labels. We extract the concatenated features from both texts and images and use a linear SVM for sentiment classification during inference. We evaluate the accuracy for the sentiment analysis and show comparable performance with the supervised methods [3, 4]. We train our model on B-T4SA dataset [3], and evaluate on B-T4SA [3] for self-supervised results and on IMDb movie reviews datasets [5] for model transfer. Results indicate that using the image-text matching as the pretext task1 boosts sentiment analysis performance with high transferability.

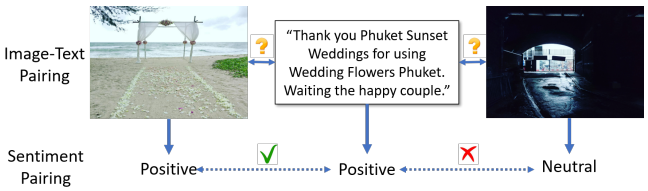


Fig. 1. An example of image-text pair and their sentiment association. When we are training a model to pair the sentence with its corresponding image, sentiment is an important attribute for identifying which image matches the given text. Models are able to extract useful sentiment information as the hint for deciding whether the given text and image are from the same scenario.

The main contributions of this work are as follows. First, we propose image-text matching as the pretext task for self-supervised sentiment analysis, where our model, Senti-ITEM, achieves a higher score with very few labels. Second, we show that the learned representation is transferable between datasets with a good ability of generalization.

2. RELATED WORKS

In this section, we introduce the recent work on sentiment analysis, image-text matching and self-supervised learning.

Sentiment Analysis focuses on finding the polarity of sentiment expressed in a given visual, audio or text message. Researchers have investigate sentiment analysis for visual

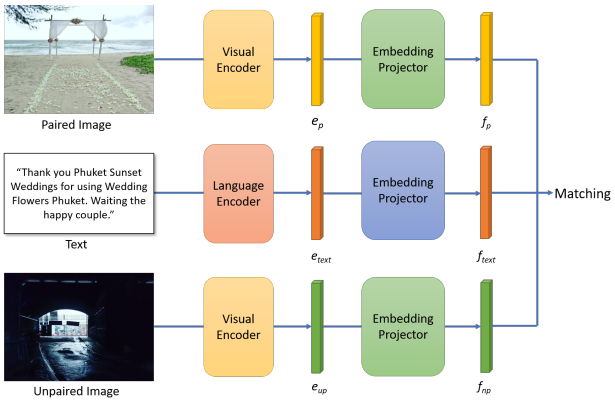


Fig. 2. Proposed architecture for training the image-text matching task for Senti-ITEM.

	B-T4SA		IMDb Movie Review	
	Train	Test	Train	Test
Images	368,585	51,000	NA	NA
Texts	309,919	49,432	25,000	25,000
Avg. Len.	12		265	
categories	3 (Pos, Neu, Neg)		2 (Pos, Neg)	
Modality	Vision + Text		Text only	
Type	Sentence		Document	
Source	Twitter		IMDb reviews	

Table 1. Statistics about B-T4SA dataset and IMDb Movie sentiment dataset.

content [6, 7], voice [8], language [9] and multimodal content [10, 11]. Most of these methods use supervised learning, where data with sentiment labels are available. Mathews *et al.* [7] apply an LSTM [12] layer after the convolution neural network for fusing the temporal information to understand the sentiments from visual data. Lopez *et al.* [11] introduce using the autoML-based fusion for multimodal sentiment understanding between visual and text input.

Image-Text Matching evaluates the similarity between the image-text pairs and identifies the matching pairs. For image-text matching, recent methods [13, 14, 15, 16] involve encoding the text and image data and measuring similarity between them. Wen *et al.* [13] apply the dual semantic relationships to pair a text with it corresponding image. Diao *et al.* [14] use the similarity between two modalities for finding their relationships. Wei *et al.* [15] introduce the general metric learning method for multimodal input matching. Wang *et al.* [16] implement a consensus-aware visual-Semantic embedding model to corporate the consensus information between modalities. Different from all these supervised methods, in this paper, we use image-text matching as the supervisory signal in self-supervised learning for sentiment analysis.

Self-supervised Learning is to train a model with a pretext task, and use the resulting encoder and its representations for the downstream tasks with minimal supervised training. There is a large and growing body of work on using self-supervised learning for image, video, and text understanding [17, 18]. By using one task as the self-supervision, researchers can train on another task with few labels or even without extra labels for supervision for such task.

3. METHOD

The architecture of the proposed method, Senti-ITEM, is shown in Figure 2. Our model consists of two parts, namely, visual and text encoders, and their corresponding embedding projectors. During training, given a text input q, we have two images, the matching image v_p and the non-matching image v_{np}. The model learns to pair the v_p with q while pushing v_{np} away from q. During inference, we use the encoder to extract features in the shared embedding space for classification with a simple classifier, *i.e.*, linear SVM.

Feature Encoders The encoders take the raw image and text as input and generate the high-level embeddings or their representations. The image encoder generates a visual embedding (feature vector) for the matching image v_p and non-matching image v_{np} to the given text as e_p and e_{np} respectively. The text encoder encodes the text input q into a language embedding e_{text}, which represents the information in the sentence or document. After encoding with the two feature encoders, we have two visual feature vectors, e_p and e_{np}, as well as a language feature vector e_{text} to evaluate the similarity between two different modalities.

Embedding Projectors The extracted features from image and text encoders are projected to a shared embedding space where there is a higher level of alignment measured through the similarity between the matching visual and text embeddings, e_p, e_{np} and e_{text}. Embedding projectors for images and texts do not share weights and generate the projected embeddings, *i.e.*, f_p, f_{np} and f_{text} respectively for two images and one text message on the shared space. During training, we use the overall similarity score S between three embeddings, where S is defined as

$$S = sim(f_{text}, f_p) - sim(f_{text}, f_{np}) \quad (1)$$

to calculate the similarity between f_p and f_{text} and between f_{np} and f_{text}, where $sim(\cdot)$ is the L-2 similarity between two embeddings. We maximize the S during training to help pushing q and v_{np} away from each other while make q and v_p closer to each other. During inference, we use the projected visual and text embedding, f_{text} and f_{img}, as features to be fed to a linear SVM classifier.

4. EXPERIMENTS AND RESULTS

In this section, we first introduce our setup for the experiments with implementation details, and followed by the quan-

Methods	Accuracy
Hyb.-T4SA FT-F [3]	49.9
Hyb.-T4SA FT-A [3]	49.1
VGG-T4SA FT-F [3]	50.6
VGG-T4SA FT-A [3]	51.3
Inf. Fusion [4]	60.4
Pretrained-Encoders (img)	47.1
Pretrained-Encoders (text)	75.7
Senti-ITEM (f_{img} only)	50.3
Senti-ITEM (f_{text} only)	76.3
Senti-ITEM	76.6

Table 2. Test accuracy for sentiment analysis on B-T4SA datset. f_{text} and f_{img} only use the feature from text and image encoder for SVM classification, while Senti-ITEM use the concatenated feature for SVM classification.

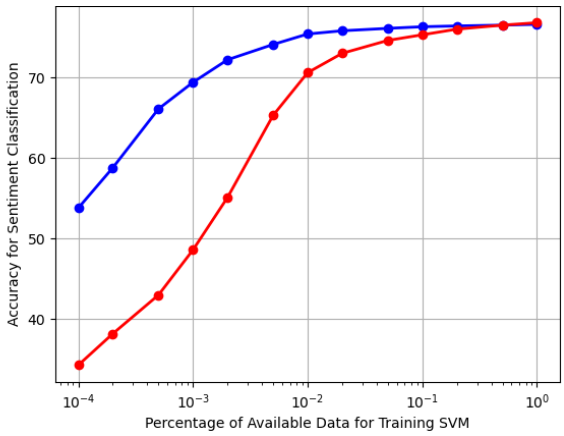


Fig. 3. Semi-supervised accuracy on B-T4SA for Senti-ITEM (blue) and the supervised MLPs (red).

titative results for analyzing the performance of using image-text matching as self-supervision for sentiment classification.

4.1. Experiment setup

Datasets We evaluate our method on B-T4SA [3] and IMDb movie reviews datasets [5]. Both of the datasets include sentiment labels allowing quantitative evaluation. B-T4SA dataset is a sentence-level Twitter sentiment analysis dataset with tweets and corresponding images. It includes three different types of sentiments, positive, neutral and negative. The IMDb movie reviews dataset is a document-level sentiment dataset collected from the IMDb reviews. It includes two different sentiments, positive and negative. All the categories for both datasets are balanced for both training and test splits. Since there is no image in the IMDb movie review dataset, we train our model on B-T4SA and evaluate on both B-T4SA and IMDb. Statistics of the two datasets are available in Table 1.

Implementation details For the language feature encoding, we use a pretrained uncased BERT [19] model to encode the language into a 768-d vector for every Twitter message. For each message q, We tokenize the whole sentences and convert them into indices with the token dictionary. Since we are using Twitter messages in the B-T4SA dataset, which are relatively short, we assume that all the messages have only one sentence. For the IMDb movie review dataset, we calculate the embeddings for every individual sentence and average the sentence embeddings for each document. For the visual encoder, we use a ResNeXt-50 [20] pretrained on ImageNet [21] to encode an image into a 2,048-d vector, extracted from the penultimate layer. Embedding projectors consist of two layers of fully connected layers projecting features into a 128-d space with ReLU as activation functions, and the sizes for hidden layers are 512.

Training and testing During training, we use the triplet loss for training the network on the B-T4SA dataset. The margin between positive and negative examples in the triplet loss is set to 0.1. We use the images from each Twitter message as 'matching image' v_p and randomly choose a non-matching image from another tweet as an unpaired image v_{np}. We train the network for 100 epochs with starting learning rate at $1e^{-4}$ and decay to half of the learning rate every 25 epochs. During inference, we first extract features for the training and test sets for both datasets with the feature projector of the language branch and then train a linear SVM on the resulting embeddings for sentiment classification. We report the overall accuracy on the test split, defined as the ratio of correct predictions to the number of all test samples. Since the numbers of examples for each category are balanced for both of the datasets we are using, we do not add extra weights for samples during the training stage.

Baseline methods Since there are no similar self-supervised methods on sentiment analysis, we compare our performance against the existing supervised sentiment analysis methods, such as [3, 4] for B-T4SA and [22, 23, 24] for the IMDb movie review dataset. We also compare with the feature e_{text} from pretrained BERT [19] and e_{img} from ResNeXt [20] without finetuned on image-text matching. We name them as 'Pretrained-Encoders (img)' and 'Pretrained-Encoders (text)' to show the impact of using image-text matching for self-supervision compared with the original performance.

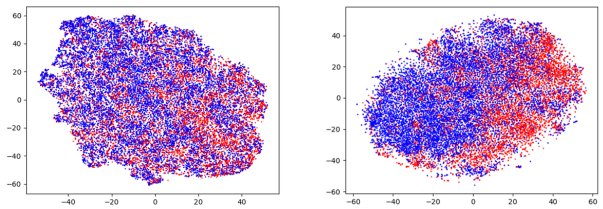
4.2. Quantitative results

For quantitative results, we show the self-supervised results and weakly supervised results with few training labels on B-T4SA dataset, as well as the transfer results on IMDb movie review dataset with the model trained on B-T4SA dataset.

Results on B-T4SA dataset The results on the B-T4SA dataset are available in Table 2. We compared Senti-ITEM

Methods	Accuracy
UnICORNN [25]	88.4
BP Transformer [22]	92.1
oh-LSTM [23]	94.1
CEN-tpc [24]	94.5
Pretrained-Encoders (text)	82.7
Senti-ITEM (f_{text})	84.2

Table 3. Accuracy on IMDb movie review test set. Senti-ITEM is the transfer results with the model trained on B-T4SA dataset, while other methods are directly trained on IMDb movie review training set with supervised scheme.



(a) (b)

Fig. 4. T-SNE visualization results for features after embedding projector for IMDb movie review test set. Red dots are positive samples, and blue dots are negative samples. (a) is the f_{text} extracted before image-text matching and (b) is extracted by Senti-ITEM after matching. (Best viewed in color.)

with several supervised methods on the first half of the table and the Pretrained-BERT, which is not finetuned with an image-text matching task. For the Pretrained-Encoders setting, the test accuracy is 75.7% and 47.1% for text and image branches, showing that there is already some helpful sentiment information in the pretrained models. After we train the network with the image-text matching task, the accuracy for Senti-ITEM is boosted to 76.3% and 50.3% for these two branches. Compared with the Pretrained-Encoders, the image-text matching task mines the useful sentiment information for both modalities. Senti-ITEM can outperform several supervised baselines (see Table 2). Compared with using the single modality f_{text} or f_{img}, the concatenated features or fusion yields the highest accuracy. Although we do not use the sentiment labels for training, the self-supervised network can extract useful features for sentiment analysis from the image-text matching task.

Semi-supervised Results on B-T4SA dataset We show the semi-supervised result for the B-T4SA dataset in Figure 3 after reducing the number of labeled training data. We compare the performance on partially labeled data with the supervised learning method, which adds a two-layer multilayer perceptron (MLP) after the concatenation of f_{img} and f_{text} and use sentiment labels for supervision during training. When the available labeled data decreases, the performance of supervised MLP sharply drops, whereas Senti-ITEM can achieve the accuracy of 53.8% with as little as 0.01% of the labeled training data. This shows that with as little as 12 examples for each category, the performance of this method is comparable to some existing supervised methods in Table 2, indicating that the representations learned by Senti-ITEM have high discriminative power.

Results on IMDb movie review dataset Besides evaluating the B-T4SA dataset, we evaluate the Senti-ITEM on IMDb movie review dataset [5]. Since there is no image in this dataset, we use the f_{text} extracted by the Senti-ITEM pretrained on the image-text matching task for the B-T4SA. Results of the first half of the table are the supervised sentiment classification accuracy, while the second half are transfer results without finetuned on IMDb movie review training split. Senti-ITEM achieves the accuracy of 84.2% for text-based sentiment analysis, which is comparable to the supervised methods (see Table 3). This shows that the features learned by self-supervised learning in Senti-ITEM are transferable across different datasets.

4.3. Qualitative results

In addition to numerical results, we also visualize the embeddings projected on the 2-D plain of the IMDb movie review dataset with sentiment categories. Figure 4 illustrates the projection results of text embeddings of the IMDb movie review dataset test set with T-SNE for dimension reduction. Figure 4 (a) shows the text features extracted from the embedding projectors for Senti-ITEM before image-text matching, while features in Figure 4 (b) are extracted through the embedding projectors from the Senti-ITEM after image-text matching. Compared with the embeddings before matching, where samples are randomly distributed on the 2-D space, text embeddings from the Senti-ITEM after matching are more linearly separable for sentiment labels. This further shows that training with the image-text matching task helps the network mine the inner sentiment information without annotations.

5. CONCLUSION

In this paper, we propose Senti-ITEM, which leverages the visual-text matching as the pretext task for sentiment analysis. While pairing the correct text with its corresponding image, the network can extract sentiment information from both image and text without sentiment labels. The trained encoder generates embeddings that are discriminative for the sentiment analysis task. The proposed approach outperforms the existing supervised models on the B-T4SA dataset. Experiments on the text-only dataset, the IMDb movie review dataset, shows that the learned representations are generalizable to text-based sentiment analysis benchmarks.

6. REFERENCES

[1] Mohammad Soleymani, David Garcia, Brendan Jou, Björn Schuller, Shih-Fu Chang, and Maja Pantic, "A survey of multimodal sentiment analysis," *Image and Vision Computing*, vol. 65, pp. 3–14, 2017.

[2] Shaha Al-Otaibi, Allulo Alnassar, Asma Alshahrani, Amany Al-Mubarak, Sara Albugami, Nada Almutiri, and Aisha Albugami, "Customer satisfaction measurement using sentiment analysis," *IJACSA*, vol. 9, no. 2, pp. 106–117, 2018.

[3] Lucia Vadicamo, Fabio Carrara, Andrea Cimino, Stefano Cresci, Felice Dell'Orletta, Fabrizio Falchi, and Maurizio Tesconi, "Cross-media learning for image sentiment analysis in the wild," in *ICCVW*, 2017, pp. 308–317.

[4] António Gaspar and Luís A Alexandre, "A multimodal approach to image sentiment analysis," in *ICIDEAL*. Springer, 2019, pp. 302–309.

[5] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts, "Learning word vectors for sentiment analysis," in *ACL-HLT*, 2011, pp. 142–150.

[6] Evangelos Sariyanidi, Hatice Gunes, and Andrea Cavallaro, "Automatic analysis of facial affect: A survey of registration, representation, and recognition," *TPAMI*, vol. 37, no. 6, pp. 1113–1133, 2014.

[7] Alexander Mathews, Lexing Xie, and Xuming He, "Senticap: Generating image descriptions with sentiments," in *AAAI*, 2016, vol. 30.

[8] Verónica Pérez-Rosas and Rada Mihalcea, "Sentiment analysis of online spoken reviews.," in *INTERSPEECH*, 2013, pp. 862–866.

[9] Asaf Beasley and Winter Mason, "Emotional states vs. emotional words in social media," in *WebSci*, 2015, pp. 1–10.

[10] Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency, "Tensor fusion network for multimodal sentiment analysis," in *EMNLP*, 2017, pp. 1103–1114.

[11] Vasco Lopes, António Gaspar, Luís A Alexandre, and João Cordeiro, "An automl-based approach to multimodal image sentiment analysis," *arXiv:2102.08092*, 2021.

[12] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] Keyu Wen, Xiaodong Gu, and Qingrong Cheng, "Learning dual semantic relations with graph attention for image-text matching," *TCSVT*, 2020.

[14] Haiwen Diao, Ying Zhang, Lin Ma, and Huchuan Lu, "Similarity reasoning and filtration for image-text matching," *arXiv:2101.01368*, 2021.

[15] Jiwei Wei, Xing Xu, Yang Yang, Yanli Ji, Zheng Wang, and Heng Tao Shen, "Universal weighting metric learning for cross-modal matching," in *CVPR*, 2020, pp. 13005–13014.

[16] Haoran Wang, Ying Zhang, Zhong Ji, Yanwei Pang, and Lin Ma, "Consensus-aware visual-semantic embedding for image-text matching," in *ECCV*. Springer, 2020, pp. 18–34.

[17] Amaia Salvador, Erhan Gundogdu, Loris Bazzani, and Michael Donoser, "Revamping cross-modal recipe retrieval with hierarchical transformers and self-supervised learning," *arXiv:2103.13061*, 2021.

[18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al., "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv:2006.07733*, 2020.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019, pp. 4171–4186.

[20] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He, "Aggregated residual transformations for deep neural networks," in *CVPR*, 2017, pp. 1492–1500.

[21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*. IEEE, 2009, pp. 248–255.

[22] Zihao Ye, Qipeng Guo, Quan Gan, Xipeng Qiu, and Zheng Zhang, "Bp-transformer: Modelling long-range context via binary partitioning," *arXiv:1911.04070*, 2019.

[23] Rie Johnson and Tong Zhang, "Supervised and semi-supervised text categorization using lstm for region embeddings," in *ICML*. PMLR, 2016, pp. 526–534.

[24] Maruan Al-Shedivat, Avinava Dubey, and Eric Xing, "Contextual explanation networks," *JMLR*, vol. 21, no. 194, pp. 1–44, 2020.

[25] T Konstantin Rusch and Siddhartha Mishra, "Unicornn: A recurrent model for learning very long time dependencies," *arXiv:2103.05487*, 2021.