Multi-Modal Representation Learning with Self-Adaptive Threshold for Commodity Verification

Chenchen Han[0000−0002−3330−5308]⋆ and Heng Jia[0000−0001−8062−9455]⋆

Zhejiang University, Zhejiang, China
hanchenchen@zju.edu.cn jiaheng.dlut@gmail.com

Abstract. In this paper, we propose a method to identify identical commodities. In e-commerce scenarios, commodities are usually described by both images and text. By definition, identical commodities are those that have identical key attributes and are cognitively identical to consumers. There are two main challenges: 1) The extraction and fusion of multi-modal representation. 2) The ability to verify identical commodities by comparing the similarity between representations and a threshold. To address the above problems, we propose an end-to-end multi-modal representation learning method with self-adaptive threshold. We use a dual-stream network to extract multi-modal commodity embeddings and threshold embeddings separately and then concatenate them to obtain commodity representation. Our method is able to adaptively adjust the threshold according to different commodities while maintaining the indexability of the commodity representation space. We experimentally validate the advantages of self-adaptive threshold and the effectiveness of multimodal representation fusion. Besides, our method achieves third place with an F1 score of 0.8936 on the second task of the CCKS-2022 Knowledge Graph Evaluation for Digital Commerce Competition. Code and pretrained models are available at https://github.com/hanchenchen/CCKS2022-track2-solution.

Keywords: Multi-modal representation · Self-adaptive threshold · CCKS-2022 competition

1 Introduction

We aims to identify identical commodities based on representation learning. Given a pair of commodities, we extract their representations and calculate the similarity between representations. Then we judge whether the pair is identical by comparing the similarity and threshold. In the second task of the CCKS-2022 Knowledge Graph Evaluation for Digital Commerce Competition, the commodity pair data is from the recall results of actual online models and manually labeled, where most of the negative pairs are similar but some key attributes do not match.

⋆ Equal contribution. Listing order is random.

The traditional identical commodity verification methods usually adopt manually adjusted thresholds. There are some disadvantages of such a method.

1) **Inter-dataset adaptation problem**. Since data distribution usually varies between datasets, the corresponding representation distribution will be different as well. The threshold determined on one dataset may be hard to achieve comparable results on another, which affects the generalization of the model. It is necessary to manually adjust the threshold, which is laborious and burdensome.

2) **Intra-dataset adaptation problem**. Since the commodity pairs are usually similar, there representations often crowded together in the representation space. A slight fluctuation of the threshold may affect the performance much. Moreover, it is unwise to use the same threshold for different kinds of commodities.

3) **Model optimization problem**. Due to the high similarity of commodities, their similarity scores are usually higher than 0. However, the existing loss functions (e.g., binary cross entropy loss) are usually centered at 0. Consequently, the model is difficult to be optimized. Besides, it may destroy the representation space to force pushing the representations of similar but non-identical commodities away.

To mitigate the above problems, we propose an end-to-end multi-modal representation learning method with Self-Adaptive Threshold (SAT). We use a dual-stream network to extract multi-modal commodity embeddings and threshold embeddings separately and then concatenate them to obtain commodity representation. Our method can adaptively adjust the threshold according to different commodities, thus reducing the burden and drawbacks of manually adjusting the threshold. The dual-stream network optimizes the commodity representation distribution bidirectionally by either the commodity stream or the threshold stream, which results in a better distribution of representations. Therefore, it is less likely to force pushing away the representations of similar but different commodities. Moreover, with our self-adaptive threshold, the similarity of representations is basically centered at 0. While maintaining the indexability of the commodity representation space, the model is easier to be optimized and the representations are more robust (more details in Section 3.3).

Our main contributions are as follows:

1) We analyze the possible problems in the traditional commodity verification approach and then propose a multi-modal representation approach with SAT to learn the threshold adaptively. Our approach reduces the burden of adjusting thresholds and enhances the generalization and robustness of the representations.

2) We do not do special processing for the inputs (e.g. no detector), and the whole network is trained end-to-end so that other methods can be easily integrated.

3) We experimentally validate the advantages of the self-adaptive threshold and the effectiveness of our multi-modal representation fusion. Our method achieves an F1 score of 0.8936 and takes third place on the second task of the CCKS-2022 Knowledge Graph Evaluation for Digital Commerce Competition.

Fig. 1. Multimodal representation learning with self-adaptive threshold. The area in the red box is the traditional method of calculating the similarity of a commodity pair. The score is obtained by subtracting the pre-defined threshold from the similarity. When the score is greater than zero, the commodity pair is predicted to be identical, and vice versa. The higher score, the greater probability of being an identical commodity pair. We add a threshold stream to learn self-adaptive threshold embeddings and regard the difference between the inner product of commodity embeddings and threshold embeddings as the score.

2   Method

In this section, we present SAT, a novel multi-modal representation learning method with self-adaptive threshold for commodity verification. We first detail the self-adaptive threshold in Sec. 2.1, then introduce the model architecture in Sec. 2.2 and the loss function in Sec. 2.3 finally. Fig. 1 shows the overview of the proposed method.

2.1   Self-Adaptive Threshold

Dual-Stream Embedding We propose to use a dual-stream network to extract the commodity embedding and threshold embedding. Given a commodity $\boldsymbol{x}$, we feed it to the commodity-stream $\boldsymbol{f}$, and extract commodity embedding:

$$\boldsymbol{p} = \boldsymbol{f}(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{p} \in \mathbb{R}^{d1}$ is the commodity embedding; $d1$ is the commodity embedding dimension. Correspondingly, we have a threshold-stream $\boldsymbol{g}$ to extract the threshold embedding:

$$\boldsymbol{q} = \boldsymbol{g}(\boldsymbol{x}) \tag{2}$$

where $q \in \mathbb{R}^{d2}$ is the threshold embedding; $d2$ is the threshold embedding dimension. Then we can acquire the complete embedding of commodity x by concatenation:

$$z = [p, q] \tag{3}$$

where $z \in \mathbb{R}^{d1+d2}$ is the complete embedding; $d1 + d2$ is the embedding dimension; $[\cdot, \cdot]$ represents concatenation.

Score Calculation As our method is based on representation learning, we do not have to tackle a commodity pair simultaneously. Given a commodity pair (x_1, x_2), we extract their embeddings separately:

$$\begin{aligned} z_1 &= [p_1, q_1] \\ z_2 &= [p_2, q_2] \end{aligned} \tag{4}$$

The similarity s is obtained by the inner product between commodity embeddings p_1 and p_2:

$$s = p_1 \cdot p_2 \tag{5}$$

where $\cdot$ represents the inner product between vectors. Correspondingly, we can get the self-adaptive threshold by the inner product between threshold embeddings q_1 and q_2:

$$t = q_1 \cdot q_2 \tag{6}$$

The final score is the difference between similarity s and threshold t:

$$SCORE = s - t \tag{7}$$

If the score is greater than 0, it is a pair of identical commodities, otherwise not. The higher score, the greater probability of being an identical commodity pair.

2.2   Model Architecture

We use the identical architecture for both streams, but in fact we can design different architectures. Taking threshold stream as example, we have a RoBERTa [4] to encode textual feature q^u from text u and a Swin Transformer [5] to encode visual feature q^v from image v. Then we concatenate them and project the concatenated embedding into a common embedding space by a linear layer h:

$$q = h([q^u, q^v]) \tag{8}$$

Similarly, we can choose other backbones to encode single-modality features.

Fig. 2. Self-adaptive threshold network. We use Swin Transformer [5] and RoBERTa [4] to encode image features and text features respectively. The features of different modalities are fused by a linear layer.

2.3 Loss Function

We use cross entropy loss [2] to train the model:

$$\mathcal{L} = -\log \frac{\boldsymbol{y} \exp\left(\boldsymbol{p}_1 \cdot \boldsymbol{p}_2\right) + (1 - \boldsymbol{y}) \exp\left(\boldsymbol{q}_1 \cdot \boldsymbol{q}_2\right)}{\exp\left(\boldsymbol{p}_1 \cdot \boldsymbol{p}_2\right) + \exp\left(\boldsymbol{q}_1 \cdot \boldsymbol{q}_2\right)} \tag{9}$$

where $\boldsymbol{y} \in \{0, 1\}$ is the ground-truth.

3 Experiments

3.1 Experimental Setup

Datasets The official dataset contains about 50,000 commodity pairs for training and about 20,000 commodity pairs for test[1]. The training is only conducted on the official training set. We do not use unlabeled data or external dataset during training. When dividing the training and validation sets, we remove the items that appear in the training set to ensure that the training set and validation set do not overlap. The ratio of the final training set and validation set is about 5.6:1. We resize all images to 384 x 384. For text, we take the title and the 10 most frequent attributes as input. We do not apply augmentations on either image or text data.

Implementation Details Our implementation is based on PyTorch [6] and HuggingFace [7]. We initialize the image encoder with Swin Transformer [5], pre-trained on ImageNet [1]. Text encoders are initialized from pre-trained RoBERTa [4]. We train SAT in an end-to-end manner. For all experiments, we use Adam optimizer [3] with betas [0.9, 0.999]. We train SAT for 100K steps on 2 NVIDIA

---

[1] https://tianchi.aliyun.com/competition/entrance/531956/information

A100 GPUs with a total batch size of 8, which takes about 20 hours. The initial learning rate and weight decay are 2e-6 and 1e-6 respectively. We use cosine annealing learning rate decay without warmup.

3.2   Ablation

In the ablation study, we validate the effectiveness of our method and analyze the impact of input modalities and pre-trained models. If not mentioned, hyperparameters other than the ablated factor are the same.

Effectiveness of SAT We first build a simple baseline as plotted in the red box of Fig. 1, which only have a commodity encoder. Besides, we add a Learnable Threshold (LT) to it. The threshold is learnable and the same for all commodities. As shown in Tab. 1, our SAT outperforms baseline methods by a large margin, indicating the effectiveness of SAT. Specifically, SAT brings significant F1-score improvements (i.e. +0.0620 higher than LT).

Table 1. Results of different methods on the validation set.

Method	F1-score	Precision	Recall	Accuracy
Baseline	0.7250	0.6097	**0.8940**	0.6432
LT	0.8204	0.8139	0.8270	0.8096
SAT	**0.8824**	**0.8795**	0.8853	**0.8759**

Impact of Modality We further analyze the input modalities. Tab. 2 shows the detailed comparisons. Image-only SAT achieves better performance than text-only, with a lead of 0.0612 on F1 score. Taking text and images together as input can further improve the performance. We believe that SAT can be further enhanced with other modality inputs, which is worth exploring in the future study.

Table 2. Results of SAT with different input modalities.

Text	Image	F1-score	Precision	Recall	Accuracy
✓		0.7888	0.7555	0.8251	0.7676
	✓	0.8500	0.8599	0.8403	0.8440
✓	✓	**0.8824**	**0.8795**	**0.8853**	**0.8759**

Table 3. Ablation study of pre-trained models

Pre-trained	F1-score	Precision	Recall	Accuracy
✗	0.7815	0.7606	0.8037	0.7637
✓	**0.8824**	**0.8795**	**0.8853**	**0.8759**

Impact of Pre-trained Models. We also study the impact of pre-trained models. As mentioned above, we use Swin Transformer [5] pre-trained on ImageNet-1k and ImageNet-22k and pre-trained RoBERTa [4]. In this ablation, we random initialize the Swin Transformer [5] and RoBERTa [4]. As shown in Tab. 3, we observe significant performance improvement with pre-trained models, which indicates the importance of pre-trained models.

3.3   Score Distribution

(a) LT   (b) SAT

Fig. 3. Visualization of score distribution. We show histograms of scores of LT and SAT. The density of the score is estimated by kernel density estimation. Compared to LT, the peak density of SAT is lower and farther away from the threshold.

Fig. 3 shows the score distribution of LT and SAT. As shown in Fig. 3(a), the density peak of negative pairs is high with LT. In the meanwhile, the density peak of both positive and negative pairs is near the threshold, which means there are quantities of pairs around the threshold. The higher density peak and the closer density peak to the threshold, the more susceptible to threshold changes. A slight fluctuation of the threshold may affect the performance much. In addition, LT heavily depends on the training data distribution, not conducive to model generalization. By contrast, the density curve of SAT is much smoother and much easier to more distinctive than LT as shown in Fig. 3(a). It can be seen that the density peak of SAT is lower and farther away from the threshold. This indicates that default threshold 0 is virtually an optimum. Therefore without

manually adjusting the threshold, we can distinguish positive and negative pairs by the default threshold of 0.

4   Conclusion

In this paper, we first analyzed the potential problems of traditional representation learning in the commodity verification task. Then we proposed SAT and demonstrated its effectiveness and advantages by quantitative experiments and score distribution visualization. With SAT, we obtained a representative and discriminative commodity representation space and achieved excellent performance. As future work, we would like to extend SAT to other multimodal representation learning tasks.

References

1. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
2. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT press (2016)
3. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
4. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
5. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)
6. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019), http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
7. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45. Association for Computational Linguistics, Online (Oct 2020), https://www.aclweb.org/anthology/2020.emnlp-demos.6