

Dialog-based Image Retrieval

ADL Final Report

Wei Li, Kao

Department of Information Management
National Taiwan University
Taipei, Taiwan
r06725021@ntu.edu.tw

Hai Tao, Wu

Department of Economics
National Taiwan University
Taipei, Taiwan
b04303128@ntu.edu.tw

Yen Ting, Lin

Department of Information Management
National Taiwan University
Taipei, Taiwan
b04705026@ntu.edu.tw

Kai Ting, Chang

Department of Information Management
National Taiwan University
Taipei, Taiwan
b04705043@ntu.edu.tw

ABSTRACT

In this work, we study the task of interactive image retrieval, where the goal is to return a target image given an existing image and textual data describing the desired modifications.

We allow users to provide feedback via natural language, and design the system such that retrieval results are iteratively refined. We reproduce the results from previous models and suggest improvements to different parts of the model architecture. Finally, we create a Line chatbot demonstrating the efficiency and effectiveness of our approach.

KEYWORDS

Image retrieval, Dialog, Natural language processing

1 INTRODUCTION

Product retrieval has become a popular topic in recent years due to the huge popularity and rapid advancement of E-commerce. In particular, the need for clothing image retrieval in the fashion industry is growing.

Previous approaches to image retrieval include relevance feedback, where the user provides a simple yes/no answer to whether an image is considered “relevant” [9] and relative attribute feedback, where the user provides feedback based on a fixed set of attributes such as “sportiness” or “furriness” [7]. Recently, [4] suggested the use of natural language for interactive image retrieval, and demonstrated the performance gains of their model over previous methods.

In this work, we base our model on the work of [4]. The detailed model structure and improvements will be shown in the next section. The main contributions of our work are as follows:

- We reproduce the work of [4] and refactor their code for improved modularity, readability and easier maintenance.
- We provide improvements to various parts of the model, including the model structure, the pre-training phase and the reinforcement learning approach, and empirically demonstrate the effectiveness of our model.
- We implement our model on an easily accessible Line chatbot, demonstrating its integration with real world applications.

2 APPROACH

The original model consists of three main components, the response encoder, the state tracker and the candidate generator. The three parts combine to form the dialog manager, which is trained in an end-to-end fashion. We use stimulated user in [4] to get relative natural language feedback.

The training process is split into two phases, a supervised pre-training stage followed by model-based policy improvement. In the following section, we explain each part of the model, the training process and the improvements we made. In the end, we explain how the chatbot is implemented and integrated with our model.

2.1 Response Encoder

The goal of the response encoder is to combine a given image and a feedback sentence into a joint representation. The feedback sentence describes how the current candidate image is different from the target image.

For the dialog turn at time step t , we obtain a candidate image a_t and a feedback sentence o_t . The image is first passed through a pretrained ResNet-101 network [5] σ with fixed parameters θ to obtain a vector representation then projected to dimension D .

$$x_t^{img} = \text{proj}(\sigma(a_t; \theta)) \in R^D$$

The feedback sentence o_t is one-hot-encoded and linearly projected then passed through a convolutional neural network π with parameters δ to obtain a vector representation as in [6]. With slight abuse of notation, the encoding process for the textual feedback can be described as follows:

$$x_t^{sent} = \pi(\text{proj}(o_t; \delta)) \in R^D$$

In our project, we experiment with different ways of encoding the text feedback for better representations. Specifically, we pass the raw text of each feedback sentence directly into a state-of-the-art text representation model BERT [3] and extract features from the hidden layers. The final vector representation of the text feedback is calculated by average pooling over the final hidden layer of the BERT model. However, we notice that this configuration leads to unsatisfactory performances in our experiments. A possible explanation is that the fashion domain has very specific vocabulary

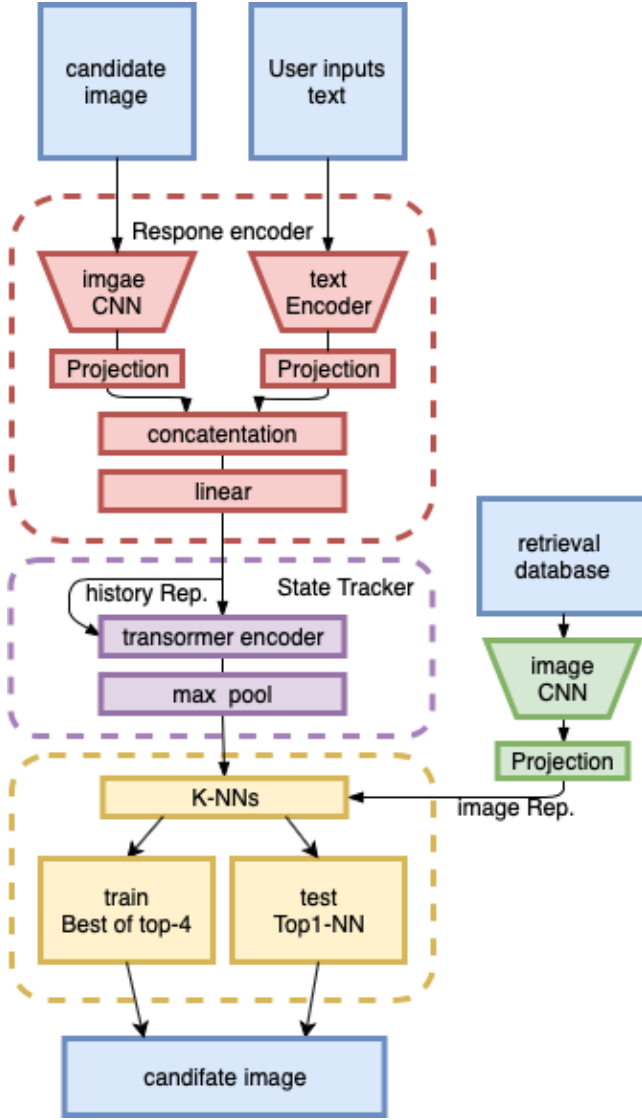


Figure 1: Model overview

and sentence structures that are significantly different from common English literature which is not captured by the pretrained BERT-Base model.

Another issue is the long training time and huge GPU memory requires for finetuning the BERT model. Since word representations obtained by the original BERT model without finetuning do not perform well, we tried using pretrained word vectors from FastText instead [2]. The final word representation for text feedback in our model is calculated by taking the average of the embedding for each word. Finally, we concatenate the embedded vectors for the image and the feedback text and pass through a linear projection layer to obtain the joint representation.

2.2 State Tracker

Our State Tracker is based on the Encoder in the Transformer model[8]. Here, the encoder maps an input sequence of history representations (x_1, \dots, x_n) , each stands for one previous dialog turn. Before the history representations are sent into the encoder, they are added with its position encoding.

The encoder is identical to the original paper, composed of a stack of $N = 3$ identical layers. Each layers contains the following sub-layers: multi-head-attention, layer norm[1], and feed forward layer. The self-attention mechanism allows the model to not only focus on the current dialog turns, but also remember previous information.

2.3 Candidate Generator

2.2 and 2.1 encode and project images and history representation of both text feedbacks and images into an embedding space and candidate generator retrieves one image from the database mainly by euclidean distance in the embedding space.

At training time, candidate generator samples from a distribution computed from top-4 nearest neighbors of s_t , history representation at turn t . The probability of choosing top- i^{th} candidates is

$$P(i) = \frac{e^{-\|x_i^{img} - s_t\|}}{\sum_{j=1}^4 e^{-\|x_j^{img} - s_t\|}}, i = 1, \dots, 4$$

At test time, candidate generator simply choose the nearest neighbors of current history representation.

2.4 Chatbot Implementation

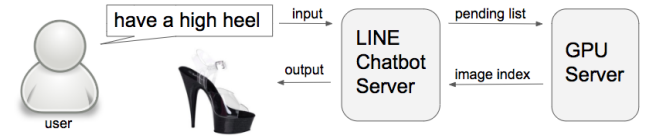


Figure 2: Chatbot Data Flow

We implemented chatbot on LINE. The user enters the target description into the LINE, and after the LINE chatbot server integrates the dialog information, it transmits the message that needs to be replied to the GPU server for inference.

Doing so allows the GPU server to focus on inferences, only replying to image index. The I/O part will be handled by the chatbot server.

3 TRAINING METHOD

3.1 Pre-training

The pre-training part is vital to this task since training the model from random initialization with policy learning method is unfeasible. Two task, triplet loss objective followed from [4] and classification task are used when pre-training.

The intuition of using triplet loss is that the proximity between target image representations x_+^{img} and history representation s_t

ADL Final Report

ensures higher ranking of target images. Loss of triplet loss is

$$\mathcal{L}^{triplet} = \mathbb{E}[\sum_{t=1}^T \max(0, \|x_+^{img} - s_t\|) - \|x_-^{img} - s_t\| + m]$$

where x_+^{img} and x_-^{img} are target image representations and random selected image representations, respectively.

We also add classification task that the history representations s_t have to predict the index of target image. The effectiveness of adding this task can be seen in 4.1.

3.2 Policy Learning

We adopt the policy improvement to our model-based policy learning. Given the current policy π and the user simulator, the value of taking an action v_t can be computed by look-ahead policy value estimation

$$\mathbb{E}[\sum_{t'=t}^T \gamma^{t'-t} r_{t'}]$$

where γ is the discounted factor and $r_{t'}$ is the reward (percentile ranking) at turn t' . We use the action that gives the highest look-ahead value in the top-4 nearest neighbor of history representation as the action for the next dialog turn. Since our user simulator is almost deterministic given same target image and history of feedbacks, we can determine the look-ahead value on single trajectory.

We refer the best action of next turn as positive and sample other 4 negative images, calculate the distribution as the same method in 2.3, and minimize the cross-entropy loss.

4 EXPERIMENTS

We used 10000 images as our training set and 4000 images as validation set. We adopted **percentile rank** as evaluation metric of our retrieval system.

Percentile rank is defined as follows:

$$1 - \frac{N(target, proposed)}{\#candidate images}$$

where $N(target, proposed)$ is number of images with closer euclidean distance to target image than the euclidean distance between target image and the image proposed by our system.

4.1 Result

We reproduced the method in [4] and got the almost identical results. In the following comparison, we would refer [4] as *previous* and our method as *ours*.

1 show the percentile rank from the first to the fifth turn of the dialog between retrieval systems and the simulated user. The result showed that our method significantly outperformed previous method even only with pre-training, and 5 explained the difference.

Table 1: Validation Result

Dialog Turn Percentile Rank(%)					
Model	1	2	3	4	5
Our Fully trained	97.12	98.93	99.09	99.15	99.15
Our Pre-trained	97.05	98.63	98.90	99.00	99.05
Previous Fully trained	95.90	98.20	98.47	98.59	98.65
Previous Pre-trained	94.98	97.80	98.16	98.25	98.28

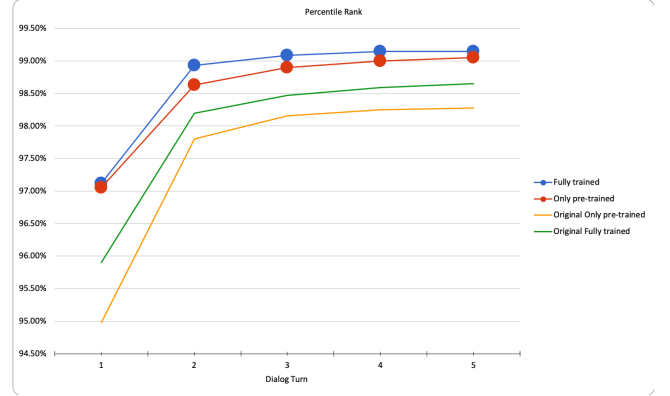


Figure 3: Chatbot Data Flow

Table 2: Training Configuration

Model	Learning Rate	Triplet Margin	Softmax Temperature
Our Fully trained	1e-5	n/a	0.1
Our Pre-trained	5e-4	0.05	n/a
Previous Fully trained	1e-3	n/a	1
Previous Pre-trained	1e-3	0.1	n/a

5 DISCUSSION

5.1 Policy Learning merely improved the system

Policy learning is aligned with our objectives, target image ranking, and reasonable learning scheme for our retrieval system containing non-differentiable K-nearest neighbor. However, the performance gain from supervised pre-training to policy learning is not significant. We come up with an explanation for the incapability of policy learning.

Reinforcement learning method is good at handling complex and dynamic environments. The environment in our case, stimulated user, is deterministic in essence, so our system would not benefit much from the strength of reinforcement learning.

5.2 Mismatched of history representation and image representation

History representation is projection of text representation and image features, while image representation is projection of image features. In the 2.3 candidate generator, the candidate for the next turn

is mainly generated by the euclidean distance of history representation and image representation, which assumes two representation exist in the same space.

Fundamentally these two representation lie in difference space. While the projection is vital for matching the spaces and can alleviate the mismatching problem, there is still room for improvement.

5.3 Comparison of Real and Simulated User

In the chatbot experiment, we found that the model can remember multiple conditional information. Taking this conversation as an example, the user specifies two conditions of high heel and clear platform in 1 and 4, and 6 "has a higher sole" can successfully produce pictures that meet the conditions and have a higher sole.





	#1	#2	#3	#4	#5	#6	#7	#8
input	have a high heel	have a clear sole	have a clear sole and clear heels	are clear platform heels	has a higher heel	has a higher sole	has a higher sole	has a higher sole
output								

Figure 4: use case

During our user experiments, we noticed a potential issue. Since we adopted a pre-training approach to help reinforcement learning converge, the model may learn to depend too much on the relative captions generated by the user simulator. For example, if the feedback text given by a real user consists solely of descriptive words instead of relative differences, or if the vocabulary of the real user differs too much from the relative captions, the model might produce inferior results.

A possible remedy to this problem is to augment the user simulator with a richer set of vocabulary. Additionally, one can also train the user simulator on descriptive text to increase the flexibility of the model.

Table 3: Examples of simulated user feedback

are gray patterned
are black suede high heel boots
are gold strappy mid heeled sandals
are brown
are slides on mules
are red mary janes
have a higher platform and more ankle strap
are brown birkenstocks
are beige open toed pumps

6 WORK DISTRIBUTION

Each member contributed **equally** to the coding part of the project and report.

Table 4: Detail Work Distribution

Name	Student ID	Contribution
林彦廷	B04705026	Candidate Generator, Training Method
高偉立	R06725021	Response Encoder
吳海韜	B04303128	Chatbot Demo
張凱庭	B04705043	State Tracker

7 CONCLUSION

In this paper, we introduce several enhancements to interactive image retrieval based on natural language processing. For encoding feedback text, we experimented with contextualized embeddings and other state-of-the-art pretrained embeddings. To help the model remember previous information, we added a Transformer architecture to the state tracker. For the supervised pre-training task, we added a classification task to directly predict the target image with history representations. Finally, we build a Line chatbot to demonstrate the results. We empirically prove the effectiveness of our method and pointed our directions for potential improvement.

We notice that the goal of increasing/decreasing euclidean distance between images in the supervised learning task does not translate directly to higher ranking percentile. Therefore, in the future, we would like to experiment with different reinforcement learning methods to more effectively interact with the simulated user. Additionally, the combined representation of text and image does not lie in the same space as the candidate images, thus improving how textual and image features are combined is also a promising research direction. Finally, we would like to extend our model to domains not limited to fashion, and incorporate different forms of multimedia such as audio or video for both input and feedback.

REFERENCES

- [1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).
- [2] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.
- [3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
- [4] Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, Gerald Tesauro, and Rogério Feris. 2018. Dialog-based interactive image retrieval. In *Advances in Neural Information Processing Systems*. 678–688.
- [5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
- [6] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).
- [7] D Parikh, A Kovashka, and Kristen Grauman. 2012. Whittlesearch: Image search with relative attribute feedback. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2973–2980.
- [8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.
- [9] Xiang Sean Zhou and Thomas S Huang. 2003. Relevance feedback in image retrieval: A comprehensive review. *Multimedia systems* 8, 6 (2003), 536–544.