ZhichunRoad at Amazon KDD Cup 2022: MultiTask Pre-Training for E-Commerce Product Search

Xuange Cui
cuixuange@jd.com
JD.com
Beijing, China

Wei Xiong
xiongwei9@jd.com
JD.com
Beijing, China

Songlin Wang
wangsonglin3@jd.com
JD.com
Beijing, China

ABSTRACT

In this paper, we propose a robust multilingual model to improve the quality of search results. Our model not only leverage the processed class-balanced dataset, but also benefit from multitask pre-training that leads to more general representations. In pre-training stage, we adopt mlm task, classification task and contrastive learning task to achieve considerably performance. In fine-tuning stage, we use confident learning, exponential moving average method (EMA), adversarial training (FGM) and regularized dropout strategy (R-Drop) to improve the model's generalization and robustness. Moreover, we use a multi-granular semantic unit to discover the queries and products textual metadata for enhancing the representation of the model. Our approach obtained competitive results and ranked top-8 in three tasks. We release the source code and pre-trained models associated with this work[1].

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**.

KEYWORDS

search relevance, e-commerce, semantic matching, multilingual

1 INTRODUCTION

With the rapid growth of e-Commerce, online product search has emerged as a popular and effective paradigm for customers to find desired products and engage in online shopping [7, 9, 11]. It is very challenging to accurately find and display relevant products. This is because the customer queries are ambiguous and implicit [12]. For example, many users search for "iPhone" to find and purchase an "iPhone charger". However, the traditional binary classification model is difficult to clearly characterize this relationship. The Amazon KDD Cup 2022 presents a novel multilingual dataset [17] across English, Japanese and Spanish, and consists of three different subtasks to evaluate the model's abilities of ranking and classifying.

In this paper, our contributions can be summarized as follows: 1) Data Augmentation. We use the translation model to convert Spanish to English for expanding the dataset. Through splitting the complement and irrelevant product text information, we could get a bigger dataset with balanced labels. We use confident learning [14, 15] to find the potential label errors and remove ~4% data from the training dataset. 2) MultiTask Pre-training. In pre-training stage, we use MLM task, classification task and contrastive learning task for improving the model's performance. 3) In fine-tuning stage, we use a multi-granular semantic unit to discover the queries and products textual metadata for enhancing the representation of the model. And we observe that exponential moving average method(EMA) [6], adversarial training(FGM) [5] and regularized dropout strategy(R-Drop) [10] could improve the model's generalization and robustness.

Our team participated in all tasks, and achieved considerably performance gain over the baseline solution. Specifically, our approach ranked 5th in task1, ranked 7th in task2 and ranked 8th in task3.

SubTask	Train Dataset	Test dataset	Languages
Task1	781K	48K	Spanish
Task2	1834K	277K	& English
Task3	1834K	277K	& Japanese

Table 1: The statistics of datasets.

2 BACKGROUND

The Amazon KDD Cup 2022 [17] provides three subtasks. The task1 consists of a query-product ranking task aimed at ranking the results list. The Normalized Discounted Cumulative Gain(nDCG) [18] will be used to evaluate the model's abilities of ranking.

The task2 and task3 are classification tasks which require the model to classify the query/product pairs into correct categories. These tasks are designed to test the model's ability of classifying. The micro-F1 [16] will be used as an evaluation metric. Moreover, the task2 consists of a multi-class product classification task aimed at classifying each product as being an Exact, Substitute, Complement, or Irrelevant match for the query. The task3 will measure the model's abilities of identifying the substitute products in the list of results for a given query.

The statistics of the corpus are shown in Table 1. In this challenge, the organizers provide two different versions of the data set. One for task 1 which is reduced version in terms of number of examples and ones for tasks 2 and 3 which is a larger [17]. It is noted that the reduced version of the data set has more difficult samples. Our team participated in all subtasks, and the next section will introduce an overview of our system.

3 SYSTEM OVERVIEW

3.1 Multi-Task Pre-Training

We compare several pre-trained multilingual language models from the XTREME Leaderboard[2], and then we use the "microsoft/infoxlm-large[3]" as text encoder.

[1]https://github.com/cuixuage/KDDCup2022-ESCI

[2]https://sites.research.google/xtreme

[3]https://huggingface.co/microsoft/infoxlm-large

The InfoXLM$_{large}$ model [1] containing 94 languages and pre-trained with CCNet dataset, and has the same configurations of XLM-R [2] and a shared vocabulary size of 250002. Figure 1 shows a high-level overview of our proposed pretext tasks.

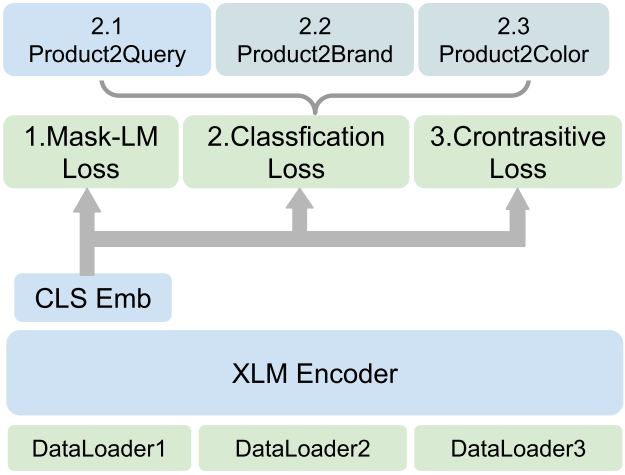

Figure 1: A schematic overview of our novel pre-training tasks. These tasks encourage the encoded representations to be more general.

MLM Task, is widely used for learning text representations [3]. MLM trains a model to predict a random sample of input tokens that have been replaced by a [MASK] placeholder in a multi-class setting over the entire vocabulary [20]. We adopt MLM-Task on the multilingual product-catalogue dataset.

Classification Task, contains three classification subtasks. One of them is Product2Query-Task, this task is a binary classification task. Based on the Poisson distribution, a piece of text is intercepted from commodity text information as the faked query. The Parameters passed to the Poisson distribution and more details can be found in appendix A.1. Product2Brand-Task and Product2Color-Task are multi-class classification that using product text information to predict the brand and the color of current item.

Contrastive Learning Task, is mainly inspired by SimCSE [4] and EsimCSE [19]. During training, each data point is trained to find out its counterpart among $(N-1)$ from in-batch negative samples and the queue of data samples. The samples in the queue are progressively replaced.

$$-\log \frac{e^{\mathrm{sim}(\mathbf{h}_i,\mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}\left(\mathbf{h}_i,\mathbf{h}_j^+\right)/\tau} + \sum_{q=1}^{Q} e^{\mathrm{sim}\left(\mathbf{h}_i,\mathbf{h}_q^+\right)/\tau}} \tag{1}$$

The h_* is the sentence representation, where h_i and h_i^+ are semantically related. The h_q^+ denotes a sentence embedding in the momentum-updated queue. And the Q is the size of the queue, $sim(h1, h2)$ is the cosine similarity scores of sentence representations, τ is a temperature hyperparameter. In the end, we average the all N Li losses to calculate the contrastive loss $\mathcal{L}_{\mathrm{con}}$.

Algorithm 1: Training a MultiTask model.

Input: DataSet $\mathcal{D} = \{(x, y, z)_i\}_{i=1}^{|\mathcal{D}|}$
Initialize model parameters Θ randomly ;
Model trainer T that takes batches of training data as input to optimize the model parameters Θ ;
Set the max number of epoch: $epoch_{\max}$;
for *epoch in* $1, 2, ..., epoch_{\max}$ **do**
  Shuffle $\mathcal{D}$ by mixing data from different tasks ;
  for $\mathcal{B}$ *in* $\mathcal{D}$ **do**
    // $\mathcal{B}$ is a mini-batch of pre-training task ;
    Compute loss : $L(\Theta)$;
    1. $L(\Theta)$ = Mask LM Loss ;
    2. $L(\Theta)$ += Classification Loss ;
    3. $L(\Theta)$ += Contrastive Learning Loss ;
    Optimize the model using $L(\Theta)$;
  end
end
Output: Pre-trained Model Θ

3.2 Fine-Tuning Methods

After pre-training, we remove the classifiers for pre-training multi-task and concatenate some embeddings with an extra MLP classifier. The embeddings consist of three sets of representations. One of them is done by concatenating the queries' 3-gram mean-pooling, bullet points' 3-gram mean-pooling and descriptions' 3-gram mean-pooling embeddings. The others consist of country embedding, brand embedding and color embedding, as shown in Figure 2.

Exponential Moving Average Our model uses EMA [6] to smooth the trained parameters. Evaluations that use averaged parameters sometimes produce significantly better results than the final trained values. Formally, we define the smoothed variables and trained variables as θ_s and θ_t, EMA decay weight as: η. After each training step, we update θ_s by:

$$\theta_s \leftarrow \eta\theta_s + (1-\eta)\theta_t \tag{2}$$

Adversarial Training Recently, adversarial attack has been widely applied in computer vision and natural language processing [5, 8, 13, 21]. Many works use it during fine-tuning, we explore the influence of adversarial training strategies and compare the FGSM, PGD, FREELB and SMART methods. The adversarial attack works by augmenting the input with a small perturbation that maximizes the adversarial loss:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\Delta x \in \Omega} L(x + \Delta x, y; \theta) \right] \tag{3}$$

where the $\mathcal{D}$ is dataset, x is input, y is the gold label, θ is the model parameters, $L(x, y; \theta)$ is the loss function and Δx is the perturbation. In our experiments, we adopt FGSM method in all tasks which based on the actual performances.

R-Drop is proved to be an effective regularization method based on dropout, by minimizing the KL-divergence of the output distributions of every two sub-models generated via dropout in model training.

$$\mathcal{L}_{KL} = \alpha \cdot \left[\mathcal{D}_{KL}\left(Logit_1, Logit_2\right) + \mathcal{D}_{KL}\left(Logit_2, Logit_1\right) \right] \tag{4}$$

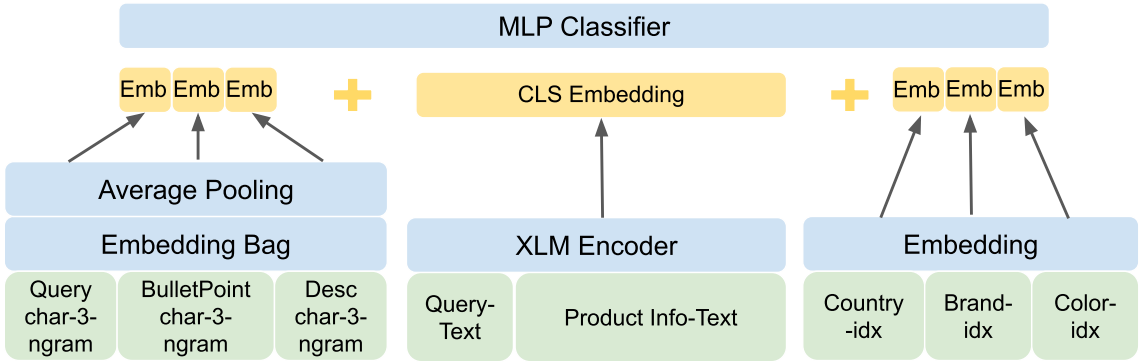

Figure 2: In fine-tuning stage, we concatenate the multi-granular semantic units, the [CLS] embedding from XLM encoder and the IDs' embeddings.

We use the origin logits of model's output as $Logit_1$, and the logits after adversarial attack as $Logit_2$.

Embedding Mixup is widely used data augmentation method through linearly interpolating inputs and modeling targets of random samples. We use the contextual embedding vector of [CLS] and the corresponding label to generate synthetic examples for training. Such training has been shown to act as an effective model regularization strategy for text classification task. In conclusion, we present the self-supervised multitask pre-training tasks and the several fine-tuning methods for improving the models' generalization and robustness.

4 EXPERIMENTS

4.1 Settings

We use InfoXLM$_{large}$ as the text encoder, the EMA decay weight is set to 0.999. And our learning rate is set to 1e-5 with warm-up ratio over 10%, batch size is 32 and gradient clip norm threshold is set to 1. In pre-training stage, the maximum number of epochs was set to 10. And in the fine-tuning stage, the maximum number of epochs was set to 5. During adversarial training, we set ε to 1.0 in FGM that means calculate only one step in the adversarial attack. We find that the dataset has imbalanced label and use some data processing steps. Through splitting the complement and irrelevant product text information, we could get more pairs which have the same label and get a more balanced dataset. We use confident learning to find the potential label errors and remove ~4% data from the training dataset. As presented in appendix A.1, the median of Spanish and English queries is 14 which satisfies the Poisson distribution of μ is set to 4. And the median of the Japanese query is 31 which satisfies the Poisson distribution with μ is set to 8.

4.2 Main Results

Our approach achieved considerably performance gain over the baseline solution, and ranked top-8 in three tasks. The main results are shown in Table 2. In task1, we calculated the mean of all model outputs as the final ranking results. In task2 and task3, we almost used the same network structure except the number of neurons in the classifier. Finally, Our approach ranked 5th, 7th and 8th, respectively.

SubTask	Model	Metric	Ranking
task1	6 large models	ndcg=0.9025	5th
task2	only 1 large model	micro f1=0.8194	7th
task3	only 1 large model	micro f1=0.8686	8th

Table 2: Performance of our approach on the private leaderboard. In task1, we used six InfoXLM$_{large}$ models that fine-tuned by different datasets or methods. In task2 and task3, we used only one InfoXLM$_{large}$ model with the same network structure, as shown in Figure 2.

Pre-Training Task	CV-MLM Loss	CV-Micro F1
Mask LM	1.966	74.97
+Product2Query	1.969	75.05
++Product2Brand	1.978	75.08
+++Contrastive Learning	2.047	75.08

Table 3: The effect of different pre-training tasks and keep accumulating from top to bottom. We report the cross validation MLM-Loss and Micro-F1 Score $\times$ 100 in the task2 setting.

4.3 Ablation Studies

We investigate the impact of adopting different pre-training task in the task2 setting. In Table 3, we show the Mask-LM losses after 5 epochs of pre-training and Micro-F1 scores after 2 epochs of fine-tuning. We find that the Product2Query task achieves an 0.008 improvement compared to the baseline, and the contrastive learning task doesn't get a significant gain.

As shown in Table 4, we compare several loss functions, and we adopt Poly1 loss function in task2 and task3 which based on the actual performances. We observe that the Focal loss function and GHM loss function have worse performance than the cross-entropy loss function in the task2 setting.

In this subsection, we explore several methods for further improving the model's performance in fine-tuning stage. As presented in Table 5, we adopt all of these methods to improve the model's generalization and robustness. We observe that the exponential moving

Classification Loss	CV-Micro F1
CE Loss	75.08
Focal Loss	74.73
GHM Loss	74.85
Poly1 Loss	75.21

Table 4: The effect of different losses in the task2 setting. We report the cross validation Micro-F1 Score × 100.

Methods	CV-Micro F1
+EMA	75.19
++FGM	75.30
+++R-Drop	75.43
++++Embedding Mixup	75.43

Table 5: The effect of different strategies and keep accumulating from top to bottom. We report the cross validation Micro-F1 Score × 100 in the task2 setting.

Confident Learning	CV-Metric
with-in-task1	NDCG, +0.005
with-in-task2	Micro-F1, -0.003
with-in-task3	Micro-F1, -0.002

Table 6: The effect of removing 4% noisy labels.

average method(EMA), adversarial training(FGM) and regularized dropout strategy(R-Drop) could improve the model's generalization and robustness. But the Embedding Mixup strategy doesn't get a significant gain.

As shown in Table 7, we consider using smaller datasets with removing ~4% noisy labels. We used the smaller dataset to achieve an 0.005 improvement in task1, but we get worse results in tash2 and task3. It could be explained that since task1 contains more difficult samples, the manually annotated data contains more label errors.

5 CONCLUSION AND FUTURE WORK

In this work, we provide an overview of the combined approach to improve the quality of search results. We use data augmentation, multitask pretraining strategy and several fine-tuning methods to achieve considerably performance. Specifically, we use mlm task, classification task and contrastive learning task in pre-training stage. And we use exponential moving average method(EMA), adversarial training(FGM) and regularized dropout strategy(R-Drop) to improve the model's generalization and robustness in fine-tuning stage. Moreover, we use a multi-granular semantic unit to discover the queries and products textual metadata for enhancing the representation of the model. Future work of our system includes: 1) Comparing with other pre-trained language models, such as deberta$_{large}$. 2) Using other training strategies, such as self-distillation.

REFERENCES

[1] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2020. InfoXLM: An Information-Theoretic Framework for Cross-Lingual Language Model Pre-Training. *CoRR* abs/2007.07834 (2020). arXiv:2007.07834 https://arxiv.org/abs/2007.07834

[2] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised Cross-lingual Representation Learning at Scale. *CoRR* abs/1911.02116 (2019). arXiv:1911.02116 http://arxiv.org/abs/1911.02116

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[4] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*.

[5] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. arXiv:1412.6572 [stat.ML]

[6] Seng Hansun. 2013. A new approach of moving average method in time series analysis. In *2013 Conference on New Media Studies (CoNMedia)*. 1–4. https://doi.org/10.1109/CoNMedia.2013.6708545

[7] Rahul Radhakrishnan Iyer, Rohan Kohli, and Shrimai Prabhumoye. 2020. Modeling Product Search Relevance in e-Commerce. *CoRR* abs/2001.04980 (2020). arXiv:2001.04980 https://arxiv.org/abs/2001.04980

[8] Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. 2020. SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 2177–2190. https://doi.org/10.18653/v1/2020.acl-main.197

[9] Sen Li, Fuyu Lv, Taiwei Jin, Guli Lin, Keping Yang, Xiaoyi Zeng, Xiao-Ming Wu, and Qianli Ma. 2021. Embedding-based Product Retrieval in Taobao Search. *CoRR* abs/2106.09297 (2021). arXiv:2106.09297 https://arxiv.org/abs/2106.09297

[10] Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-Drop: Regularized Dropout for Neural Networks. *CoRR* abs/2106.14448 (2021). arXiv:2106.14448 https://arxiv.org/abs/2106.14448

[11] Yiqun Liu, Kaushik Rangadurai, Yunzhong He, Siddarth Malreddy, Xunlong Gui, Xiaoyi Liu, and Fedor Borisyuk. 2021. Que2Search: Fast and Accurate Query and Document Understanding for Search at Facebook. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (2021).

[12] Hanqing Lu, Youna Hu, Tong Zhao, Tony Wu, Yiwei Song, and Bing Yin. 2021. Graph-based Multilingual Product Retrieval in E-commerce Search. *CoRR* abs/2105.02978 (2021). arXiv:2105.02978 https://arxiv.org/abs/2105.02978

[13] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2019. Towards Deep Learning Models Resistant to Adversarial Attacks. arXiv:1706.06083 [stat.ML]

[14] Curtis G. Northcutt, Lu Jiang, and Isaac L. Chuang. 2021. Confident Learning: Estimating Uncertainty in Dataset Labels. *Journal of Artificial Intelligence Research (JAIR)* 70 (2021), 1373–1411.

[15] Curtis G. Northcutt, Tailin Wu, and Isaac L. Chuang. 2017. Learning with Confident Examples: Rank Pruning for Robust Classification with Noisy Labels. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence* (Sydney, Australia) *(UAI'17)*. AUAI Press, 10 pages. http://auai.org/uai2017/proceedings/papers/35.pdf

[16] Juri Opitz and Sebastian Burst. 2019. Macro F1 and Macro F1. *CoRR* abs/1911.03347 (2019). arXiv:1911.03347 http://arxiv.org/abs/1911.03347

[17] Chandan K. Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping Queries Dataset: A Large-Scale ESCI Benchmark for Improving Product Search. arXiv:2206.06588

[18] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Tie-Yan Liu, and Wei Chen. 2013. A Theoretical Analysis of NDCG Type Ranking Measures. *CoRR* abs/1304.6480 (2013). arXiv:1304.6480 http://arxiv.org/abs/1304.6480

[19] Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2021. ESimCSE: Enhanced Sample Building Method for Contrastive Learning of Unsupervised Sentence Embedding. *CoRR* abs/2109.04380 (2021). arXiv:2109.04380 https://arxiv.org/abs/2109.04380

[20] Atsuki Yamaguchi, George Chrysostomou, Katerina Margatina, and Nikolaos Aletras. 2021. Frustratingly Simple Pretraining Alternatives to Masked Language Modeling. *CoRR* abs/2109.01819 (2021). arXiv:2109.01819 https://arxiv.org/abs/2109.01819

[21] Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2020. FreeLB: Enhanced Adversarial Training for Natural Language Understanding. In *International Conference on Learning Representations*. https://openreview.net/forum?id=BygzbyHFvB

Methods	CV-Micro F1
Random$^{\diamondsuit}$	-
Word2vec$^{\clubsuit}$	85.33
Freeze$^{\heartsuit}$	85.29

Table 7: The performance of different initialization methods of the multi-granular semantic unit. We report the cross validation Micro-F1 Score $\times$ 100 in the task3 setting.

A APPENDIX

A.1 Poisson Distribution

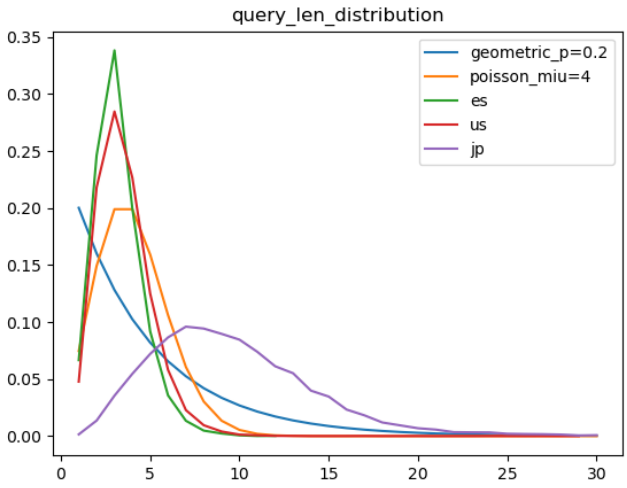

Figure 3: The length distribution of queries in different languages.

As presented in Figure 3, the median of Spanish and English queries is 14 which satisfies the Poisson distribution of μ is set to 4. And the median of the Japanese query is 31 which satisfies the Poisson distribution with μ is set to 8.

A.2 EmbeddingBag Initialization

The multi-granular semantic unit implemented by Embedding-Bag[4]. As presented in Table 7, the way of random initialization converges slowly, so we don't record the final result. And when the Embedding-Bag is initialized by Word2vec, our approach obtain the best performance.

[4]https://pytorch.org/docs/stable/generated/torch.nn.EmbeddingBag.html