Multiclass Product Classification Based On Multilingual Model and LightGBM (Team:Uni)

Peng Zhang*
Zhejiang University
China

Linghan Zheng*
Ant Group
China

Ruiqing Yan*
CNIC, CAS; BIPT
China

Changyu Li*
University of Electronic Science and Technology of China
China

Rui Hu
Ant Group, China
China

Sheng Zhou
Zhejiang University
China

Jinrong Jiang; Lian Zhao
CNIC, CAS; University of Chinese Academy of Sciences
China

Qianjin Guo; Qiang Liu
AAI, BIPT
China

ABSTRACT

With the rapid development of machine learning and data mining, the retrieving and ranking systems of e-commerce have significantly improve the searching quality and users experience. In this paper, we propose an effective solution for multiclass product classification which has won the third place in the KDD Cup'22 challenges task2 (Multiclass Product Classification). In our solution, we considered the multiclass product classification as a sentence classification problem, which is one of classic natural language understanding problem. We propose a method with cross-lingual Bert models and ensemble strategy trained with purely text information of queries and products. Based on the proposed method, we achieve a micro-F1 score of 0.8273 in the final private dataset. It is worth noting that in task3 (Product Subtitle Identification), our team also achieves the 3rd place with a micro-F1 score of 0.8754. We claim that we have not used any external data (The information of product pages crawled by web crawlers and images of products), any data leakage from task1 and even the training dataset of task1.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**; • **Applied computing** → **Document searching**; *Online shopping*.

KEYWORDS

neural networks, relevance learning, nature language processing, online shopping

ACM Reference Format:
Peng Zhang, Linghan Zheng, Ruiqing Yan, Changyu Li, Rui Hu, Sheng Zhou, Jinrong Jiang; Lian Zhao, and Qianjin Guo; Qiang Liu. 2022. Multiclass Product Classification Based On Multilingual Model and LightGBM (Team:Uni). In *KDD Cup 2022 Workshop: ESCI Challenge for Improving Product Search, August 17, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 4 pages.

*Equal contribution

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
KDDCup '22, August 17, 2022, Washington, DC, USA
© 2022 Copyright held by the owner/author(s).

1 INTRODUCTION

This paper describes our solution for multiclass product classification, , which takes the 3rd place in the final leaderboard of KDDCup'22 competition. This competition requires the completion of query and product semantic matching tasks on the "Shopping Queries Dataset"[15], which contains different Amazon search queries and results in English, Spanish, and Japanese. The dataset contains about 130 thousand unique queries and 2.6 million manually labeled (query, product) relevance judgments.

The basic idea of our solution is to perform ensemble learning through LightGBM after the language models are stacked separately. Meanwhile, some effective features constructed by feature engineering are also introduced into LightGBM during ensemble learning. More specifically, we choose the multilingual pre-training model InfoXLM and the English pre-training model DeBERTaV3 as the language models in our solution since the "Shopping Queries Dataset" contains multiple languages and the proportion of English reaches 69.4%. Due to the large gap in structure and information between the two models we selected above, it is very beneficial to obtain a higher score when the results are fused.

When fusing the two language models for different language ranges, we introduce query and product country location information to enhance the integration. During the analysis on the bad case, we found that the language model will classify the ESCI labels of the same query sample into same result (especially exact). We optimize the above situation when using LightGBM ensemble learning.

2 RELATED WORK

Recently, deep learning-based methods have been successfully applied to various relevant learning. These methods are roughly divided into two categories[9]. One is to embed two input sentences separately with deep learning-based methods and computes the similarity between these two embeddings. The other is to concatenate two input sentences as a single input to the neural network and the final relevance score is computed in an end-to-end manner.

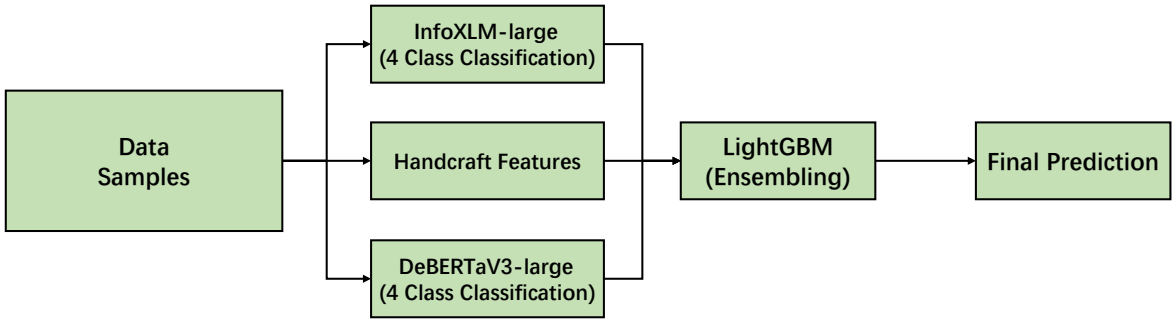

Figure 1: Overall structure of our proposed method.

In the first category of methods, DSSM [8] builds a neural network with two sub-models that learns the representation of query and document independently. The representation is also known as embedding vectors, and the relevance score of a given query-document pair is simply the similarity (*e.g.*, inner product) of the two output vectors. Followed by these 2-stage methods, later works [1, 11] focus on obtaining better representations, or retrieving efficiently [3]. These methods have high retrieval speed since the static sentences could be pre-computed and cached, and ANN (approximate near-neighbor) algorithm helps quickly retrieve the required similar sentences. However, these models suffer from the lack of information interaction between the two towers and eventually resulting in lower performance. Therefore, our method is not based on a two-stage manner.

In the second category, the end-to-end relevance learning task is also one of the pre-training tasks in BERT [4], which is known as the Next Sentence Prediction (NSP) task. In this task, two sentences will be concatenated as a single input. NLP models have to predict whether two sentences appear consecutively or not based on this fusion input. Although the end-to-end method consumes more time, it also has better performance which is the criteria for the competition.

3 MODEL ARCHITECTURE

This section introduces the detailed design of the proposed method. As shown in Figure 1, our method consists of four parts: InfoXLM, DeBERTaV3, Training Strategy and Ensemble strategy. First, we introduce the pre-trained cross-lingual language model InfoXML [2]. Secondly, we introduce the English pre-trained model DeBERTaV3 [6, 7]. Thirdly, we introduce our training strategy. Forthly, we introduce several training strategies used in InfoXML and DeBERTaV3. Finally, we proposed a new ensemble strategy based on LightGBM [10].

3.1 InfoXLM

The dataset is multilingual and contains English, Japanese, and Spanish. Samples in the dataset are composed of queries and products, which could be in different languages. This requires the model to recognize different languages. Therefore, we use the InfoXLM [2] model in our strategy, which is one of the most powerful cross-lingual pre-trained models. Different from previous transformer based language models that are only trained with a single language, InfoXLM is trained with both monolingual and parallel corpora, which contains more than 100 languages. Besides, previous State-of-the-art cross-lingual pre-trained models are typically trained using multilingual masked language modeling (MLM), alongside Translation Language Modeling (TLM). InfoXLM combines these two methods with a new cross-lingual pre-training task named cross-lingual contrast (XLCO). The final loss of cross-lingual pre-training in InfoXLM is defined as the weighted sum of these three objectives.

3.2 DeBERTaV3

The competition dataset includes millions <query, product> sample pairs in 3 different languages: English, Japanese, and Spanish and they are at a ratio of 7:1.5:1.5. Therefore, a well pre-trained English model could improve the overall performance prominently.

DeBERTaV3[6, 7] is a powerful pre-trained English language model. Different from Bert [5], DeBERTa [7] is equipped with disentangled attention mechanism and an enhanced mask decoder. Firstly, the word embedding in disentangled attention mechanism is represented by two separate embedding vectors: one for the content embedding and the other for the position embedding. Secondly, similar to Bert, DeBERTa is trained with masked language modeling (MLM), which means randomly masking some words in a sequence with a special masking token and directing the model to fill these masks with its original words. The necessary condition for the MLM task is that the model learns the contents and position information of the sentence. However, the disentangled attention mechanism only considers the contents and relative positions while ignoring the absolute positions of these words, which could degrade the performance. Therefore, DeBERTa implements an enhanced mask decoder that adds absolute position information of the words at the decoder layer. Finally, DeBERTaV3 replaces MLM tasks with replaced token detection (RTD) and proposed a new embedding sharing method. In our method, DeBERTaV3 is also used as a feature extractor. Similar as InfoXLM, The final output of DeBERTaV3 is the four class prediction probability.

3.3 Training Strategy

Samples in the dataset are listed as <Query, Product> pairs. A query is a short phrase that marks the search request, and Product contains the following information: product_id, product_title, product_description, product_bullet_point, product_brand, product_color_name, product_locale. Sample pairs will be pre-processed as follows:

$$\begin{aligned} Input = {} & [CLS] + Query + [SEP] + product\_id + [SEP] \\ & + product\_brand + [SEP] \\ & + product\_color + [SEP] + product\_title\_name + [SEP] \\ & + product\_bullet\_point + [SEP] \\ & + product\_description + [SEP] \end{aligned}$$

where $[CLS]$ is a special classifier token that is used when doing sequence classification, $[SEP]$ is also a special token that is used to denote the split position between sentences. InfoXLM will be used as a feature extractor to map the *input* into a dense embedding x, and then a fully connected layer will transformer x into the four class probability.

We have tried many finetuning strategies to improve Bert Model performance. Significantly effective strategies are adversarial training, contrastive learning, hard sample mining, and product_id mining method. We extend the adversarial training method FGM [13] to the text domain by applying perturbations to word embeddings in Bert Model. We use the contrastive learning method Rdrop [12], which forces the output distributions of different sub-models generated by dropout to be consistent with each other. And in each mini-batch, we sort the loss computed in the forward propagation phase from all samples and select the top 85% of them as hard samples [16]. Therefore we only compute the gradient from the hard samples in the backward propagation phase, which means we ignore the easy samples that are less helpful to the Bert model while training. We found that the product_id of each product has rich information on product categories. Without external data, we put the first six letters of product_id in the product text. Once the first letter of product_id is a numerical value and less than eight, we inserted "book" in the front of the product text to guide Bert models to know the category.

3.4 Stacking

Due to different model architectures, pretrained tasks, pre-training corpus, and finetuning strategies, our infoXML model and DeBERTaV3 show different advantages in samples from different regions. Ensembling these models with a simple vote strategy can not show the advantages of each model in different regional samples. So we use the stacking strategy to solve the above problems. We take the models' probabilities, **query_locale**, and other handcraft features as the input features. Then we use a well-known Gradient Boosting Decision Tree Model named LightGBM [10] to balance the result of each model and make the final decision. The stacking strategy improves the performance by 1% compared to the just voting strategy.

Specifically, we take the predicted probabilities from infoXML and DeBERTaV3 as the base features(during the training stage, the probabilities are predicted using a k-fold cross-validation style, and during the testing stage, the probabilities are the blending result from the k-fold models). Then **query_locale** is taken into consideration to tell the model which country this sample comes from. We found that samples from Japan were mixed with English. So we identify the real language of query and product side features and then use them as features.

Inspired by [14], we designed a set of match scores based on query term and product term, e.g. Jaccard similarity between **query** and **product_title**. These term-level match scores slightly improve the performance by 0.1%.

To further improve the performance, we conduct a bad case analysis on the offline validation dataset. We find that our models often predict all samples under the same query into the same class. We hope that models can correct this problem in the stacking stage, so we build a series of aggregation features of the predicted result of all samples under the same query. These features significantly improve the result of our model by 0.4%.

3.5 Performance optimization

Due to the limitation of online computing resources, and meeting the official requirements, we optimized the performance of the inference process of our solution. In our scheme, significant computational demands occur during the inference process of the language model. To optimize the inference performance of the language model, we build a computational graph using mixed precision for inference. At the same time, we performed operator fusion for all operators except matrix multiplication, which reduced GPU kernel scheduling and memory reading and writing. It is optimized for frequently called "hot operations" such as GELU activation functions, layer regularization, and softmax. Finally, on a machine with an 8-core CPU, 32G RAM, and an Nvidia Tesla V100 PCIe 16GB GPU, the inference time on the test set using InfoXLM-Large (or DeBERTaV3-Large) was reduced from more than 30 minutes to less than 5 minutes. It also ensures that the 4-bit effective precision of the inference result is not affected.

4 RESULTS

Based on the online F1 scores, here are some experimental results of this competition. First of all, we were not in a hurry to try knowledge distillation. We trained multiple models based on multi-fold cross-validation in order to reduce the variance of model predictions. Five Infoxlm-large models in cross-validation without any fine-tuning strategies reach 0.814 in the public test F1 score. Our highest score is the ensemble model of the Bert models, artificial features, and LightGBM models. We noticed that the improvement based on ensemble between LightGBM models is very limited, but the Bert models and LightGBMs model can bring a huge improvement of 1%, which we believe is due to the huge difference between the various Bert models. With knowledge distillation, the performance of a single model grows about 1%, but there is no benefit to the ensemble result nor is adversarial training tricks. We use 4 multi-folds models (18 models in total)to get the best F1 score of 0.8281 in the public test F1 score. Due to the limitation of model

Language Models	Input Length	Training Strategy	Model Ensembling (KFold)	Handcraft Features	Model Ensembling Strategy	Online F1 score
InfoXLM-large	180	-	5	-	Average	0.8142(public)
InfoXLM-large	128	Rdrop, FGM	5	-	Average	0.8163(public)
InfoXLM-large + DeBERTaV3	128	Hard-Sample-Mining, Rdrop, FGM	4	-	Average	0.8184(public)
InfoXLM-large + DeBERTaV3	128	Hard-Sample-Mining, Rdrop, FGM	4	Add	LightGBM Blending	0.8281(public)
InfoXLM-large + DeBERTaV3	128	Hard-Sample-Mining, Rdrop, FGM	4	Add	LightGBM Blending	0.8274(private)

Table 1: Performance comparison with different models

inference time, we submit 9 models to get 0.8274 in the private test F1 score.

5 CONCLUSION

In this paper, we propose a method based on cross-encoder Bert model and LightGBM for multiclass product classification, in which multiple fine-tuning strategies are executed in the language model and many useful handcraft features are used in LightGBM. Our method is based on pure text information of the query and products, which is provided in the training dataset of the KDDCup'22 challenge task2. Our method is very to the point and better serves the reality of product searching and sorting problem without any leakage. At the same time, we achieved the 3rd place in KDDCup'22 challenge task2 and task3.

ACKNOWLEDGMENTS

We thank both KDD organizers as well as Amazon for holding such a great competition. This study is supported by the National Natural Science Foundation of China (Grant No.41931183). The numerical calculation in this work were carried out on the SunRising-1 computing platform. This study is also supported by National Natural Science Foundation of China (Grant No: 62106221)

REFERENCES

[1] Qingyao Ai, Daniel N. Hill, S. V. N. Vishwanathan, and W. Bruce Croft. 2019. A zero attention model for personalized product search. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*. Wenwu Zhu, Dacheng Tao, Xueqi Cheng, Peng Cui, Elke A. Rundensteiner, David Carmel, Qi He, and Jeffrey Xu Yu, (Eds.) ACM, 379–388. DOI: 10.1145/3357384.3357980.

[2] Zewen Chi et al. 2021. InfoXLM: an information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Online, (June 2021), 3576–3588. DOI: 10.18653/v1/2021.naacl-main.280.

[3] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15-19, 2016*. Shilad Sen, Werner Geyer, Jill Freyne, and Pablo Castells, (Eds.) ACM, 191–198. DOI: 10.1145/2959100.2959190.

[4] 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Jill Burstein, Christy Doran, and Thamar Solorio, (Eds.) Association for Computational Linguistics, 4171–4186. DOI: 10.18653/v1/n19-1423.

[5] 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Jill Burstein, Christy Doran, and Thamar Solorio, (Eds.) Association for Computational Linguistics, 4171–4186. DOI: 10.18653/v1/n19-1423.

[6] Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. (2021). arXiv: 2111.09543 [cs.CL].

[7] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*. https://openreview.net/forum?id=XPZIaotutsD.

[8] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry P. Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*. Qi He, Arun Iyengar, Wolfgang Nejdl, Jian Pei, and Rajeev Rastogi, (Eds.) ACM, 2333–2338. DOI: 10.1145/2505515.2505665.

[9] Yunjiang Jiang, Yue Shang, Rui Li, Wen-Yun Yang, Guoyu Tang, Chaoyi Ma, Yun Xiao, and Eric Zhao. 2019. A unified neural network approach to e-commerce relevance learning. In *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data* (DLP-KDD '19) Article 10. Association for Computing Machinery, Anchorage, Alaska, 7 pages. ISBN: 9781450367837. DOI: 10.1145/3326937.3341259.

[10] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, (Eds.), 3146–3154. https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html.

[11] 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. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*. Feida Zhu, Beng Chin Ooi, and Chunyan Miao, (Eds.) ACM, 3181–3189. DOI: 10.1145/3447548.3467101.

[12] 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. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, (Eds.), 10890–10905. https://proceedings.neurips.cc/paper/2021/hash/5a66b9200f29ac3fa0ae244cc2a51b39-Abstract.html.

[13] Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=r1X3g2%5C_xl.

[14] David Rau and Jaap Kamps. 2022. How different are pre-trained transformers for text ranking? In *European Conference on Information Retrieval*. Springer, 207–214.

[15] 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. (2022). arXiv: 2206.06588.

[16] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. 2016. Joint face detection and alignment using multi-task cascaded convolutional networks. *CoRR*, abs/1604.02878. http://arxiv.org/abs/1604.02878 arXiv: 1604.02878.