KDD CUP 2022 MULTICLASS PRODUCT CLASSIFICATION: TEAM MetaSoul SOLUTION

Zhichao Feng, Jiawei Lu, Junwei Cheng,
Ke Hou, Kaiyuan Li, Pengfei Wang, Yadong Zhu

Introduction

In this paper, we focus on the second task of the KDD Cup, a multi-class product classification task which aims to predict the relationship between the query and the product retrieved for this query using the query statement and the product metadata.

Our solution achieves an overall Micro-F1 of 0.8207, which wins fifth place in the final leaderboard1.

Experiment

In this part, we list the results of our experiments. Remarkably, we performed our models with different tricks on the sampled dataset and full dataset respectively when working offline.

model	micro-F1(%)
mDeBERTa	74.05
mDeBERTa + EMA	74.15
mDeBERTa + SWA	74.18
mDeBERTa + AWP	74.99
mDeBERTa + Multi-Sample Dropout	74.26
mDeBERTa + Label Smoothing	74.16

Table 1: Summary of the sampled dataset, including the number of unique queries, the number of judgements, and the average number of judgements per query.

model	SM	MPE
mDeBERTa + AWP	76.37	76.41
DeBERTa-Large(only English)	77.3	77.4
XLM-RoBERTa + multidropout	75.55	75.76

Table 2: Offline micro-F1(%) of models on full dataset. SM represents single model and MPE represents model parameters ensemble.

model	Public micro-F1(%)	Private micro-F1(%)
mDeBERTa + AWP	81.48	81.72
Ensemble	81.82	82.07

Table 3: Online performance of models on full dataset. Ensemble consist of the three models in Table 2.

Method

In this part, we list the methods we used. The structure of our model is shown in Fig 1.

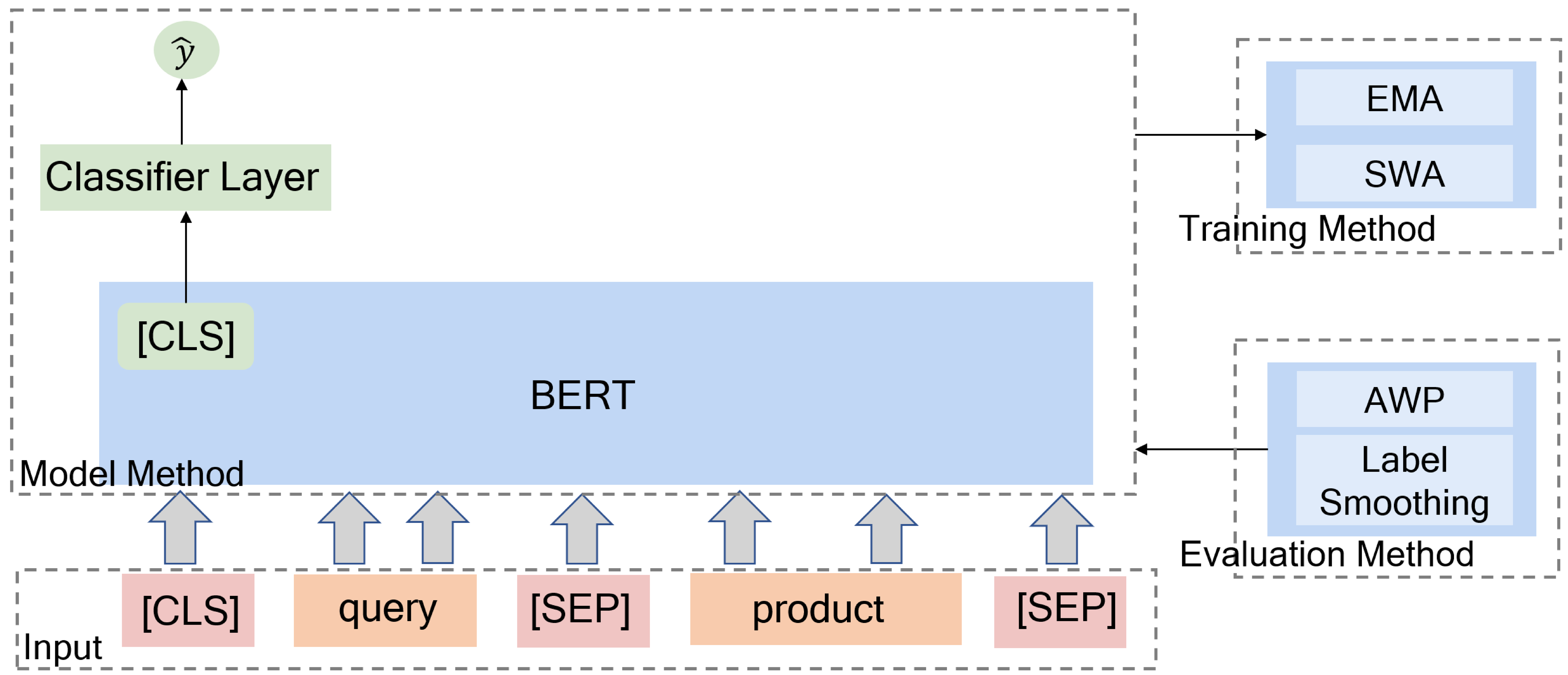

Figure 1: The structure of our model. Our backbone is a single-tower model. On the right of the figure are some of the tricks we tried. The output of classifier layer is probability of four classes.

Firstly, we perform word tokenization on the preprocessed query and product text. Then we put them into the bert to get embedding information. At this part, we also use several kinds of bert and we'll introduce them later. Finally we use the [CLS] token of bert with tricks to classify the input data. We will introduce our models from 3 followeing methods.

Model Method:

We use the structure in Fig.1 to classify product-query pairs. As for the bert part, we mainly use mDeBERTa, XLM-RoBERTa, DeBERTa-Large to encode product and query information. Multi-dropout is also used to accelerate training and improve model's generalization.

Training Method:

We deploy adversial training to perturb the input or model parameters to construct adversarial examples and improve the robustness of model while training. We also use label smoothing method through soft label to address the problem of overfitting and over confidence.

Evaluation Method:

To get a better model, we use Exponential Moving Averages (EMA) and Stochastic Weight Averaging(SWA) method to average model parameters of different training period. Without affecting training speed, we only need to train one model and store two models during training.

Post Processing

Inspired by EMA, we consider it is beneficial to maintain moving averages of the trained model parameters. When most of the model parameters are generally consistent within a time window, it indicates that the parameters around this place are relatively confident. So we use the weighted averages of last several checkpoint model parameters instead of the last trained values and experiment proves it efficient.

Paper and code:

https://github.com/guijiql/kddcup2022