Solution for Dacon Korean NLI Task Challenge

Junha Park, Hongsun Jang, Jinho Jeong

March 1, 2022

Abstract

Our approach to Dacon Korean NLI Task Challenge, which includes fine-tuning of KoELECTRA, ROBERTa pretrained model achieved accuracy of 0.88115 on predicting KLUE Benchmark Dataset. Custom loss function, data augmentation with back-translation, soft-voting ensemble techniques are applied to enhance classification performance.

1 Task Description

NLI, Natural Language Inference Task is known as a type a Natural Langue Processing task of classifying relationship between two sentences. Set of two sentences(Premise, Hypothesis) are labeled as 'entailment, neutral, contradiction', according to the relationship of two sentences.

Dacon Korean NLI Task Challenge benchmarks KLUE Dataset , which includes 24,998 train data, 3,000 dev data, and 3,000 test data [1]. Hypothesis and gold standard labels are created through crowdsourcing, when premise sentence is presented. Premise sentences are collected through web crawling, from sources which includes WIKIPEDIA, WIKITREE, WIKINEWS, POLICYNEWS, AIRBNB, NSMC. Proportion and distribution of data from each data sources are quite uniform.

2 Baseline Models

KoELECTRA baseline model is exploited for various experiments, due to its high efficiency[2]. Variations that enhances performance of KoELECTRA baseline model are also applied to ROBERTa baseline model. Pretrained KoELECTRA-base model, implemented as HuggingFace API are fine-tuned with custom dataset. ROBERTa pretrained baseline model, which is implemented as HuggingFace API are fine-tuned with custom dataset to obtain robust and powerful performance.

3 Variations

3.1 Data Augmentation : Back-translation

While fine-tuning pre-trained model, representation collapse occurs during fine-tuning and leads to overfitting[3]. When this phenomena occurs, both validation accuracy and loss increases. Thus, data augmentation is time-consuming but effective approach, especially dealing with overfitting issue. Back-translation, EDA(Easy Data Augmentation) are well-known data augmentation techniques for NLP Task. While back-translation was effective, easy data augmentation didn't enhanced the performance of baseline models. Back-translation is performed with drawling, exploiting papago web application via Selenium and BeautifulSoup. It took about 44 hours to translate to English and back-translate into Korean for 10000 sets of (premise, hypothesis).

Additionally, 3000 KLUE dev sets are also concatenated to train set after data leakage test. Collectively, 37,998 train sets are prepared.

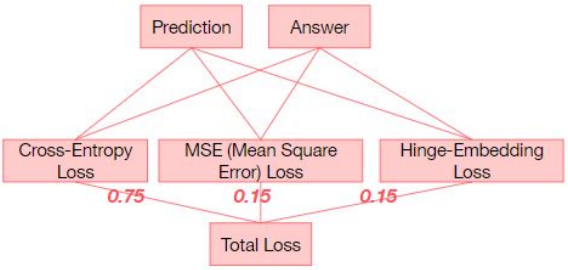
3.2 Custom Model

ROBERTa, KoELECTRA pre-trained model should be fine-tuned to classification downstream task. Therefore, fully connected layers and dropout layers were stacked, following the pre-trained model. Details are presented in this repository.

3.3 Custom Loss Function

Custom loss function is adopted to deal with overfitting issue. While fine-tuning a pretrained network into classification downstream task, phenomenon which validation loss and accuracy both increases are observed. We assumed that overfitting problem is caused by representation collapse while fine-tuning is on progress. Aghajanyan et al.[3] suggested novel loss function(RXF loss) to tackle this problem. Thus, our postulate was that customized loss function might soothe the problem that was described above. Angular loss[4], supervised contrastive loss[5] are also proposed. However, for simple implementation, we adopted ensemble of MSE, hinge embedding loss, and cross entropy loss, which was proposed in Hajiabadi et al.[6] Weights for ensemble were manually optimized through grid search, observing validation loss tendency.

$$\mathcal{L} = \mathcal{L}_{ce} \times 0.75 + \mathcal{L}_{mse} \times 0.15 + \mathcal{L}_{he} \times 0.15 \tag{1}$$



3.4 Ensemble

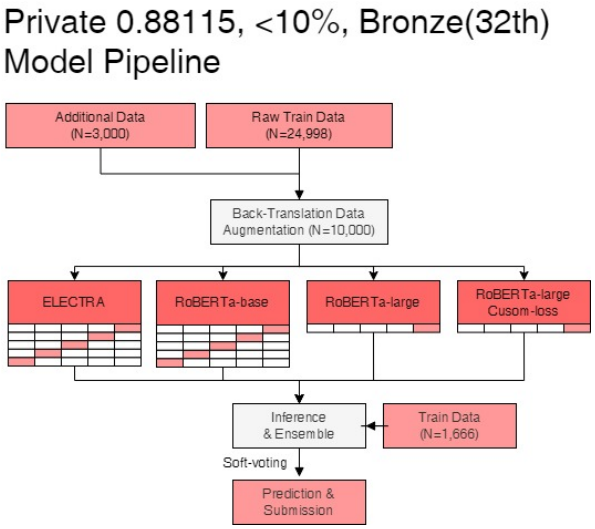
Ensemble of different models, especially with cross-validation helps to acquire robust prediction. Soft-voting ensemble are applied, to softmax probability of each models, instead of raw logits.

$$\mathcal{S}_{ensemble} = \mathcal{S}_{RL} \times 0.25 + S_{RLCL} \times 0.25 + \mathcal{S}_{RB5CV} \times 0.25 + \mathcal{S}_{EB5CV} \times 0.25 \tag{2}$$

(RL : Roberta-Large, RLCL : Roberta-Large Custom-Loss, RB5CV : Roberta-Base 5 cross-validation ensemble, EB5CV : ELECTRA-Base 5 cross-validation ensemble)

4 Final Pipeline

Final Pipeline for best submission is same as following.



5 Result

The table below describes the performance of vanilla model, without any augmentation, loss function customization, and ensemble. Learning rate are benchmarked from BERT paper and epoch is limited to 10 due to restricted GPU environment. Experiment result is presented as following.

Model	Score
KoELECTRA	0.866
ROBERTa base	0.861
ROBERTa large	0.875
Ensemble	0.886

References

[1] S. Park, J. Moon, S. Kim, W. I. Cho, J. Han, J. Park, C. Song, J. Kim, Y. Song, T. Oh, *et al.*, "Klue: Korean language understanding evaluation," *arXiv preprint arXiv:2105.09680*, 2021.

[2] J. Park, "Koelectra: Pretrained electra model for korean," *GitHub repository*, 2020.

[3] A. Aghajanyan, A. Shrivastava, A. Gupta, N. Goyal, L. Zettlemoyer, and S. Gupta, "Better fine-tuning by reducing representational collapse," *arXiv preprint arXiv:2008.03156*, 2020.

[4] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin, "Deep metric learning with angular loss," in *Proceedings of the IEEE international conference on computer vision*, pp. 2593–2601, 2017.

[5] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18661–18673, 2020.

[6] H. Hajiabadi, D. Molla-Aliod, R. Monsefi, and H. S. Yazdi, "Combination of loss functions for deep text classification," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 4, pp. 751–761, 2020.