Supervised Classification Task for Relation Classification

Introduction

For the task of relation classification, the significance of model selection and vocabulary size is foreseeable. This assignment aims to develop a classifier that is capable of automatically identifying entity relationships based on sentence-level input and the indexes of the two entities. This assignment also empowers students to do experiments on the effects of using different models with contextual and non-contextual embedding vectors. Therefore, I conducted three experimental analyses using different pre-trained models with different frameworks.
The procedure starts with dataset loading, turning textual data into numeric representations in a regular case. The following step is doing experiments using three different models with different embedding sources. The default embedding source is the pre-trained Glove with 100-dimensional word embeddings of tweets. Other available embedding sources such as FastText and Word2Vec have also been experimented with and evaluated based on their performance. Another option is to create contextual embedding using a pre-trained BERT tokenizer and pre-trained BERT model. The last step is to feed processed inputs into different language models. These experimental results will be demonstrated in tabular format.

Experiments For Different Models

This section has conducted experiments exploring the effects of changing the embedding techniques, the representation at different embedding levels, the number of embedding dimensions, and the preference of different language learning models on the variation of accuracy rate. Given that the nature of this assignment is to find the most appropriate relation categories based on two entities included in a sentence, the input vectors for the model shall comprise semantic information regarding the two entities and the sentence itself.

1. Model Comparison

As I mentioned above, the performance of three different models with different tokenizers will be examined. The first model, serving as the baseline model, is fully developed on the architecture offered in the skeleton. This pure attention-based model consists of a default tokenizer, an example classifier with a BaseEncoder and a BinRelEncoder. The default tokenizer splits sentences by whitespaces without considering issues of punctuation and capitalization. However, the selection of tokenizers significantly impacts the modeling result. Using a malfunctioning tokenizer on a dataset would generate a poor-quality vocabulary, which in turn would cause a skyrocketing number of out-of-vocabulary. To mitigate this problem, a series of text preprocessing would reduce the number of OOV from approximately 1,200 to 200 for an 8,000 instances dataset.
The second model is solely stored in the Bert_Classification file. The second experiment is inspired by an RBER model introduced by Wu et al. in 2019 (Wu et al., 2019). This research paper aims to develop a model that leverages the pre-trained BERT model and incorporates information extracted from the two target entities. In this experiment, the second model uses a pre-trained BERT tokenizer from the mxnet framework for dataset processing. However, using the Bert tokenizer would cause an issue of inconsistency between the real entity position and the stored entity's position. Since the BERT tokenizer relies on Byte-Pair Encoding, tokenized sentence length would vary based on cases. Therefore, the positional index of the real two entities shall vary following sentence length variation. In the next step, a pre-trained BERT model takes the indexed

results as inputs and yields a sequence encoding vector and a CLS encoding vector. The CLS encoding representation and the vector representations for each of the two target entities are concatenated horizontally. Then, the concatenated vector is fed as input into one fully connected layer and one dense layer. The third experiment is a replicate of the research paper. The only difference is replacing the proposed RBERT model with a pre-defined RoBerta model.

	Precision	Recall	Acc	AP
Baseline - Train	0.606	0.588	0.764	0.596
Bert - Train	0.6842	0.6667	0.9998	0.999
Roberta -Train	**0.6420**	**0.6145**	**0.8435**	**0.7198**
	Precision	**Recall**	**Acc**	**AP**
Baseline - Val	0.514	0.500	0.49	0.267
Bert - Val	0.5658	0.5483	0.6448	0.4345
Roberta - Val	**0.6267**	**0.6092**	**0.8275**	**0.6938**
Best Practice	Model (Roberta); Epochs (10); Optimizer (Adam); Learning Rate (0.0001); Batch Size (36); Drop Out Rate (0.1); Sequence Length (32); Bert Contextual Embedding; Embedding Size (128); Accuracy: 0.8275 Acc on Validation Dataset			

Table 1 - Model Performance with Hyper-parameters Tuning and Random Embedding

The experiment results yielded by the pure-attention model training on GloVe embedding are recognized as baselines. The baseline model performs relatively inferior to others. One possible explanation is that the attention layer with positional encoding is insufficient to capture the essential information from a long sequence sentence. One possible solution is to combine a seq2seq model with the baseline model. The accuracy rate for the BERT model (which is 0.6448) on the validation set is noticeably higher than the baseline. Similarly, the accuracy rate of the RoBerta model is significantly higher than the previous one. One potential explanation is that BERT was undertrained. RoBerta's capability to capture essential information from longer sequences and dynamic masking pattern enables it to process long sequence sentences more effectively.

2. Error Analysis

Tables 2 and 3 summarize relation types with the top 3 lowest or highest F1-score from the third experiment. Interestingly, "Other" has the lowest recall rate of merely 0.5308 in the validation set. A potential explanation is that sentences assigned in the "Other" type may contain various relationships that could be future classified into sub-types. On the contrary, cause-effect relation is the easiest type to be classified. This result is reasonable since sentences with causal-effect are prone to include specific keywords such as "cause," "lead to," and "induce" for demonstration.

Class	Precision	Recall	F1-Score
Other	0.6965	0.5308	0.6025
Member-Collection	0.6486	0.7500	0.6956
Instrument-Agency	0.7727	0.7727	0.7727

Table 2 – Bottom 3 Relation Types ordered by F1-Score

Class	Precision	Recall	F1-Score
Cause-Effect	0.9136	0.9477	0.9304
Entity-Destination	0.9078	0.9484	0.9277
Cause-Effect-Inv	0.9230	0.9278	0.9254

Table 3 - Top 3 Relation Types ordered by F1-Score

Reference:

Wu, Shanchan, and Yifan He. "Enriching pre-trained language model with entity information for relation classification." In *Proceedings of the 28th ACM international conference on information and knowledge management*, pp. 2361-2364. 2019.