Machine Learning Engineer Nanodegree

Capstone Project Proposal

by Wasifa Chowdhury

Domain Background

Natural language processing (NLP) has grown by leaps and bounds over the past few years with the emergence of deep learning technologies. Machine learning models can now tackle complex tasks like question answering, text extraction, and sentence generation. A characteristic of natural language is that there are many different ways to convey the same information - several meanings can be contained in a single text and that the same meaning can be expressed by different texts. While humans can process the meaning of a sentence rather quickly, can we measure how well machines navigate the ambiguity of multiple senses of words and draw on context to determine relationship between sentences? If NLP can be applied between sentences, this could have profound implications for real-world applications like fact-checking, identifying fake news, and much more.

Moreover, most of the work to date on NLP has been focused mainly on English, leaving low- and medium-resource languages behind. To address this problem, Kaggle has initiated a competition, Contradictory, My Dear Watson, to challenge machine learning practitioners to build a system that automatically classifies how pairs of sentences are related from texts in fifteen different languages. The aim of this capstone project is to create a multi-class classification model using transfer learning.

My personal motivation for this project comes from the fact that I'm interested in how natural language understanding can be used to break down language barriers. Also as someone whose native language is a scarce-resource language, it motivates me more to investigate better machine learning methods for multilingual corpus.

Problem Statement

Textual entailment recognition, also known as natural language inferencing, is the task of deciding, given two text fragments, whether the meaning of one text can be inferred from another text. Textual entailment measures natural language understanding as it requires computing semantic interpretation of the text.

For premise-hypothesis sentence pairs, the goal is to predict whether a hypothesis statement is true (entailment), false (contradiction), or undetermined (neutral) given a premise sentence. Here entailment is a directional relation which means that the hypothesis must be entailed from the given premise, but the premise need not be entailed from the hypothesis.

The pair of input sentences are in fifteen different languages, including Arabic, Bulgarian, Chinese, German, Greek, English, Spanish, French, Hindi, Russian, Swahili, Thai, Turkish, Urdu, and Vietnamese. Thus the goal is to train a multilingual model that can handle multiple languages simultaneously and classify input pairs into the three classes.

Datasets and Inputs

The dataset consists of train and test files with the following format:

- train.csv: This file contains the ID, premise, hypothesis, and label, as well as the language of the text and its two-letter abbreviation
- test.csv: This file contains the ID, premise, hypothesis, language, and language abbreviation, without labels.

Here the ID column holds a unique identifier for each sample while the label column stores the classification of the relationship between the premise and hypothesis (e.g. 0 for entailment, 1 for neutral, 2 for contradiction). The train set contains 12120 data examples while the test set contains 5195 examples in the aforementioned languages.

Solution Statement

In this project, transfer learning with data augmentation will be used to train deep learning models with Transformer[1] architecture and classify the textual entailment relationship between premise-hypothesis pairs. Instead of training a separate model for every language, a single model will be trained on the multilingual corpus to handle multiple languages simultaneously. Since the training dataset is small consisting of less than 100k examples, we will fine-tune publicly available pre-trained language models that are significantly cheaper than training a model from scratch. We will use the XLM-RoBERTa model from Facebook that has been trained on 2.5TB of filtered CommonCrawl data in 100 different languages. Data augmentation in the form of back translation will also be used to boost model's performance.

Benchmark Model

Random choice: This is a naive approach where there is an equal probability for an input sample to belong to any class of the three output classes. This submission yields ~33% accuracy in the Kaggle leaderboard.

Multilingual BERT: The Multilingual BERT or M-BERT by Google is a single language model pre-trained on the concatenation of monolingual Wikipedia corpora from 104 languages. We fine-tune the M-BERT model on our training dataset to get the baseline predictions for textual entailment recognition. We extract the pretrained BERT vector representations from the *[CLS]* or classification token in the last layer and pass that as input to a linear layer with *Softmax* activation function for further training. This yields an accuracy of ~65% in the submission leaderboard.

Evaluation Metrics

Accuracy is used as the evaluation metric to quantify the performance of both the benchmark model and the solution model presented. For each sample in the test set, the goal is to predict whether a given hypothesis is related to its premise by contradiction, entailment, or whether neither of those is true (neutral). Those values map to the logical condition as:

0 == entailment

1 == neutral

2 == contradiction

Hence accuracy returns the percentage of correct predictions on the test data.

Project Design

Programming language: Python 3.6
Libraries: Keras, Tensorflow, Torch, Scikit-learn, Pandas, Huggingface/Transformers, Huggingface/Datasets
Workflow:

- Establish the baselines with random choice and M-BERT.
- Fine-tune a XLM-RoBERTa model on the training dataset and use a separate validation set to monitor early stopping and avoid overfitting.
- Use hyperparameter tuning to compare the performance of multiple trained models and save the configuration settings of the best model.
- Augment the training data with back translated inputs and train the XLM-RoBERTa model with the best configuration settings.
- Consider experimenting with different feature extraction strategies in the pretrained network (e.g. max/avg pooling, etc.).

References

1. https://papers.nips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf