Lab 07 – NLP – Computer-based testing

Trimester 1 - 2022

Table of Contents

1. Word Embedding

Open quora.ipynb and follow the instructions.

Objectives

Quora is a popular website where anyone can ask and/or answer a question. There are more than 100 millions unique visitors per month.
Like any other forum, Quora is facing a problem: toxic questions and comments.
As you can imagine, Quora teams cannot check all of the Q&A by hand. So they decided to ask the data science community to help them to perform automatically insincere questions classification.

Guidelines

This challenge was launched on Kaggle : https://www.kaggle.com/c/quora-insincere-questions-classification
Read the overall information on Kaggle. Quora provided a dataset of questions with a label, and the features are the following:

- qid: a unique identifier for each question, a hexadecimal number
- question_text: the text of the question
- target: either 1 (for insincere question) or 0

The Kaggle dataset is quite heavy and it may be too difficult for your laptops to perform the computations. Therefore, we provide you with **the train dataset** (to be sampled) and also **light word embeddings**, which you can download here

Don't look at the published kernels, in order to keep your judgement unbiased.

Here are a few steps to follow:
1. First sample the dataset to 10.000 lines otherwise your laptop might die (you may need to use sklearn.utils.resample()).
2. As usual, begin with a proper EDA.
3. Perform a nice text preprocessing.
4. Try to run a quick sentiment analysis using TextBlob.
5. Then, use a word embedding (Glove) to create your corpus and run your model.
6. Do some optimization (text preprocessing, model hyperparameters, other word embeddings if you trust your computer).
7. Optimize ++: Now, let's have a look at some published kernels and find some inspiration.
8. Bonus question : try to identify the most recurrent topics in toxic questions !
In this competition, the metric used for performance evaluation is the **F-score**.

2. Text classification

Objectives

The objective is to assign a given news headline to one of the categories "Business," "Science," "Entertainment". We use the News Aggregator Data Set given by Fabio Gasparetti in this exercise.

The tasks include:

- Text pre-processing
- Feature extraction
- Training, prediction and evaluate model.
- Hyper-parameter tunning

Guidelines

Download the data from this link and generate training data (train.txt), validation data (valid.txt), and test data (test.txt) as follows:

- Read readme.txt after extracting the downloaded zip file.
- The articles should be extracted such that the publication is one of "Reuters", "Huffington Post", "Businessweek", "Contactmusic.com", and "Daily Mail".
- The retrieved articles should be mixed at random.
- The extracted articles are distributed as follows: training data (80%), validation data (10%), and test data (10%). Then save them as the train.txt, valid.txt, and test.txt files, respectively. Each line inside a given file should contain a single instance. Each instance must include the category and article titles. Separate fields with Tab characters.
- Verify the number of occurrences in each category after producing the dataset.

Extract a collection of features from the training data, validation data, and test data, accordingly. Train.feature.txt, valid.feature.txt, and test.feature.txt are appropriate filenames for the features. Design the characteristics that will aid in the categorisation of news articles. The tokenized sequence of a news headline serves as the minimal standard for features.

Utilize the training data to calibrate the logistic regression model.

Utilize the logistic regression model. Create a program that predicts the category of a given news headline and determines the likelihood of the model's forecast.

Calculate the accuracy score of the logistic regression model using both training and test data.

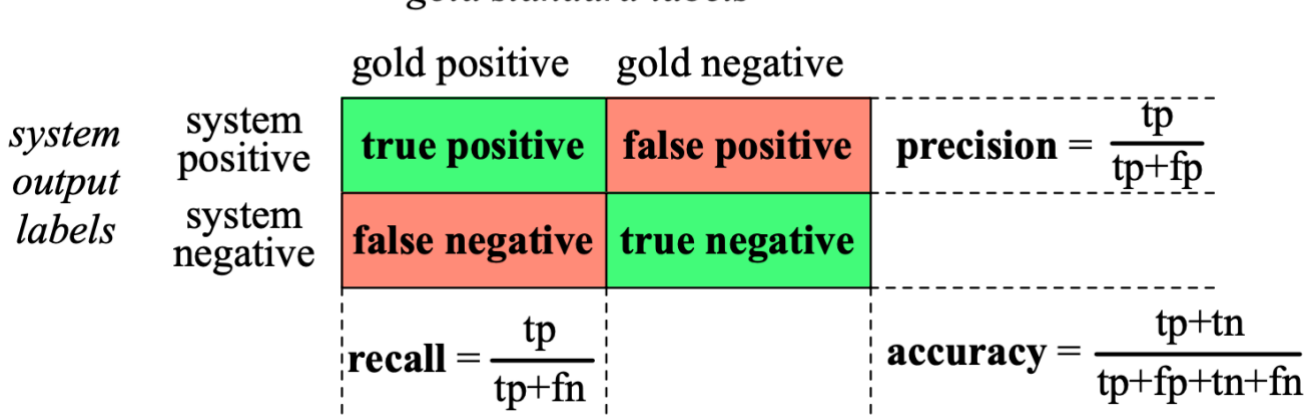


Figure 1 Evaluation metrics explanation [1]

Both the training data and the test data should be used to generate the confusion matrix of the logistic regression model.

Calculate the accuracy, recall, and F1 score for the logistic regression model. First, calculate each category's metrics. Then, summarise the score for each category using (1) micro-average and (2) macro-average.

Utilize the logistic regression. Check the feature weights and identify the top ten and bottom ten most significant features.

By adjusting the regularisation parameters during training of a logistic regression model, it is possible to regulate the degree of overfitting. Train the model using a variety of regularisation settings. Calculate the accuracy score using the training data, validation data, and test data. Create a graph with the regularisation parameter along the x-axis and the accuracy score along the y-axis.

Train the model for news classification using a variety of techniques and parameters. Search for the training techniques and settings that provide the highest level of accuracy on the validation data. Then, calculate its precision score using the test data.

3. Sentiment analysis

This exercise is about sentiment analysis and its application in real world corpus.

The tasks include:

- Text pre-processing
- Sentiment analysis

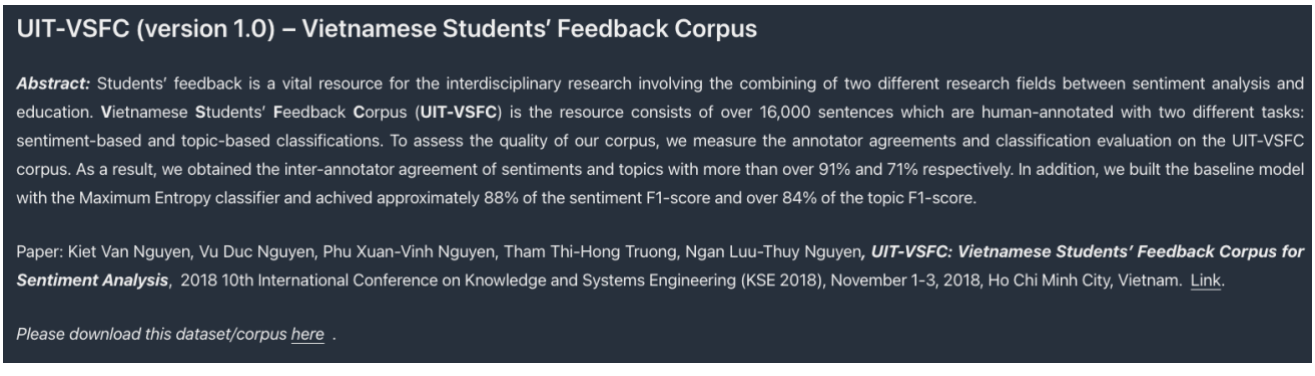
It's time to make our real NLP problem on real dataset: sentiment analysis on UIT Vietnamese Students' Feedback Corpus (UIT-VSFC) [1]
Sentiment analysis classifies a text as reflecting the positive or negative orientation (sentiment) that a writer expresses toward some object [2].

Data description: VSFC is a corpus with 16,000 sentences. Primarily, there are two types of feedback: (1) feedback given by lecturers to students in order to make them more aware of their academic strengths and weaknesses in order to improve their studies; and (2) feedback given by students to lecturers in order to enable lecturers to reflect on and improve their own teaching methods. In particular, students voice their perspectives on a range of addressed themes. Each academic semester concludes with the administration of a course survey in order to collect student feedback.
If you want to know more about this corpus, please read this paper.

Task definition: Determine whether a feedback sentence from a Vietnamese student expresses a positive, negative, or neutral/objective sentiment.

To do that, we will reuse our knowledge: we will apply all knowledge in this course on a dataset of texts.

Guidelines
Firstly, you need to access to this link and find VSFC dataset, after that, you need to download it.



This dataset consists of three folders: train, dev, and test. Each folder contains three text files: sentiments.txt, sents.txt, and topics.txt. This task requires you to focus on sentiments.txt and sents.txt and use only these files to complete it.
Sentiments.txt contains labels with positive, negative, and neutral.
Each sentence in Sents.txt is compatible with labels in sentiments.txt.

Evaluation metrics: Precision, Recall, F1-score and Accuracy.

Here are a few steps to follow:
1. Import your data into your laptop (if it is strong enough) or Google Colab.
2. Perform a nice text preprocessing.
3. Run word embedding (Glove, fasttext or any models that are effective for Vietnamese).
4. Run sentiment analysis model (train and predtict) with fine-tunning hyperparameter. You can try BERT, PhoBERT, RoBERTa, XL-NET.
5. Evaluate your result with Accuracy, Precision, Recall and F1-Score.

4. Machine translation (Optional)

This exercise is about machine translation in real world corpus.

Objectives

In this exercise, we train a neural machine translation (NMT) model by using IWSLT'15 [3] English to Vietnamese translation dataset.

The tasks include:

- Text pre-processing
- Training, prediction and evaluate model.
- Hyper-parameter tunning

Guidelines

Download the data from this link and split training data into about 10,000 sentences. Your laptop or device may be crashed if you use all more than 100,000 sentences.

Using the dataset, train the model of the machine translation system. The choice of model architecture is left to the discretion of the user (e.g., LSTM-based model, Transformer-based model).

Utilize the trained model and execute the programme to generate a German sentence based on a given English sentence.

Using the training model, compute the BLEU (Bilingual Evaluation Understudy) score on the test data.

Be careful to use beam search while translating a given text using the model. The beam width should be increased from one to one hundred. After that, plot the change in the BLEU score on the graph.

Using tools such as TensorBoard, see how the training of the machine translation model is proceeding. Important indicators include the loss and BLEU score on training data, as well as the loss and BLEU score on validation data.

Modify the architecture and/or hyperparameters of the machine translation model and search for the version that achieves the highest performance on the validation data.

REFERENCES

[1]. Jurafsky, D., & Martin, J. H. (2022). Speech and language processing (3rd ed. draft).

[2]. K. V. Nguyen, V. D. Nguyen, P. X. V. Nguyen, T. T. H. Truong and N. L. -T. Nguyen, "UIT-VSFC: Vietnamese Students' Feedback Corpus for Sentiment Analysis," 2018 10th International Conference on Knowledge and Systems Engineering (KSE), 2018, pp. 19-24, doi: 10.1109/KSE.2018.8573337.

[3]. Minh-Thang Luong and Christopher Manning. 2015. Stanford neural machine translation systems for spoken language domains. In Proceedings of the 12th International Workshop on Spoken Language Translation: Evaluation Campaign, pages 76–79, Da Nang, Vietnam.