

Assignment #6

Text Sentiment Classification with Prompt Learning

Due on Dec 20, 23:59 (11:59 pm)

Overview

- In this assignment, you will implement the sentiment classification to identify the emotional tone behind a body of text.
- In Task 1, you need to fine-tune a pre-trained language model (e.g., BERT) to predict the sentiment of given tweets.
- In Task 2, you will employ **prompts** to enable the model to perform sentiment analysis through **in-context learning**, eliminating the need for additional training.
- In Task 3, you will utilize the LM-BFF method to automatically generate the optimal template and verbalizer using the model.

Dataset

- Task1, 2: Twitter US Airline Sentiment from Kaggle
 - Twitter data was scraped from February 2015 for each major U.S. airline. Contributors were asked to first classify tweets as positive, negative, or neutral, followed by categorizing negative reasons.
 - The data is split into train, val and test
 - Category label: 0 (negative), 1 (neutral), 2 (positive)
- Task 3: SST-2
 - The sentences in the dataset are sourced from movie reviews and are designed for the task of binary sentiment analysis. Each sentence is annotated with either a positive or negative sentiment label. The data is split into train, val and test
 - Category label: 0 (negative), 1 (positive)

Dataset examples

- Task1, 2: Twitter US Airline Sentiment from Kaggle

- Predict the label of tweets

- Negative: 0
 - Neutral: 1
 - Positive: 2

text	sentiment_label
@USAirways i have been patient.. especially the first time. i will never fly with you again	0
@JetBlue y'all got prices for the low??	1
@AmericanAir thank you for responding rather quickly btw	2

- Task 3: SST-2

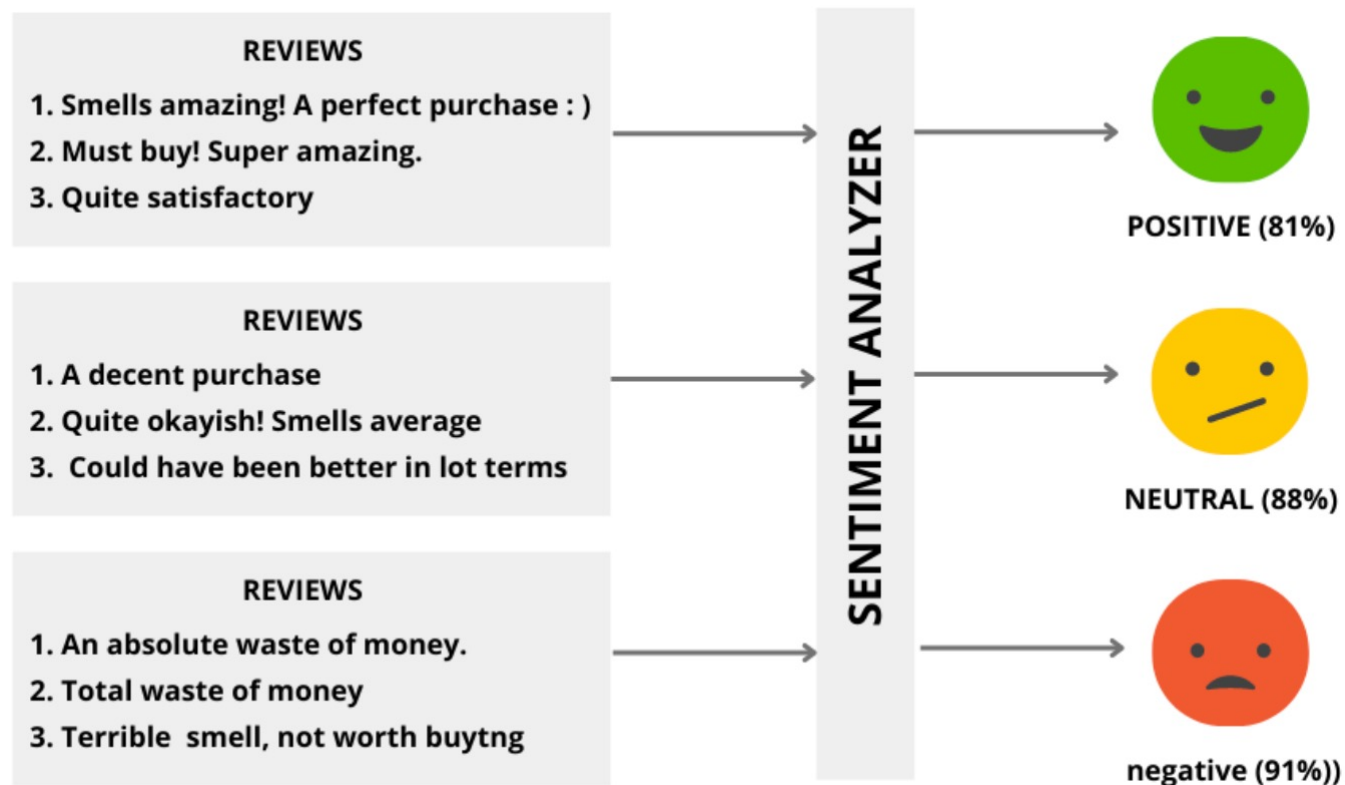
- Predict the label of movie review

- Negative: 0
 - Positive: 1

sentence	label
nothing happens , and it happens to flat chara	0
sweet and memorable film .	1

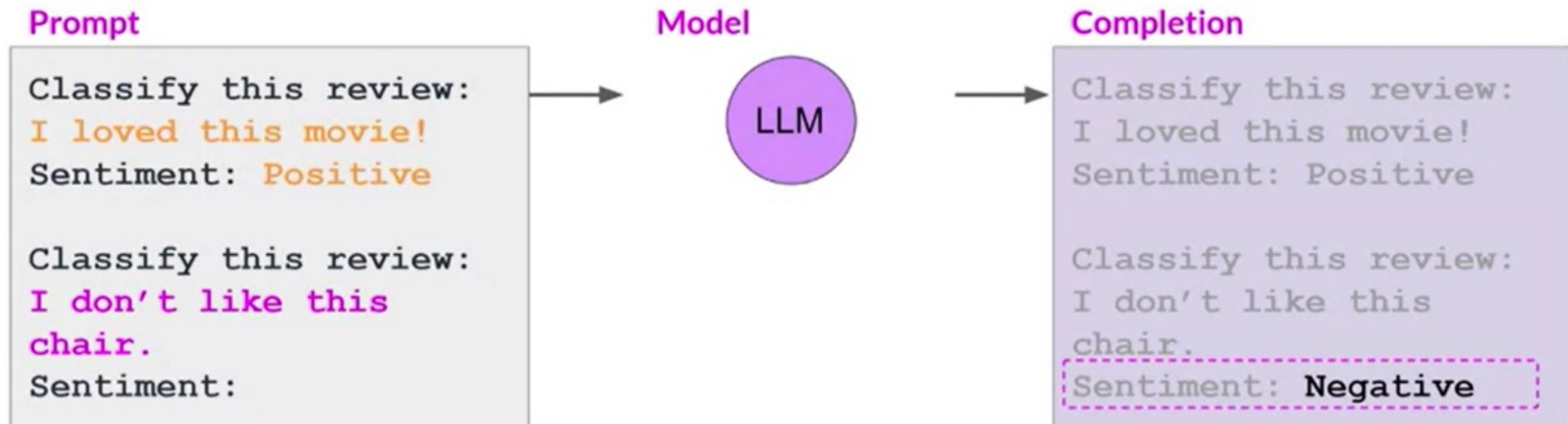
Task 1: classic sentiment classification

- Fine-tune a pre-trained model of your choice (e.g., BERT) on the given data
 - [PyTorch-transformers pre-trained models](#)



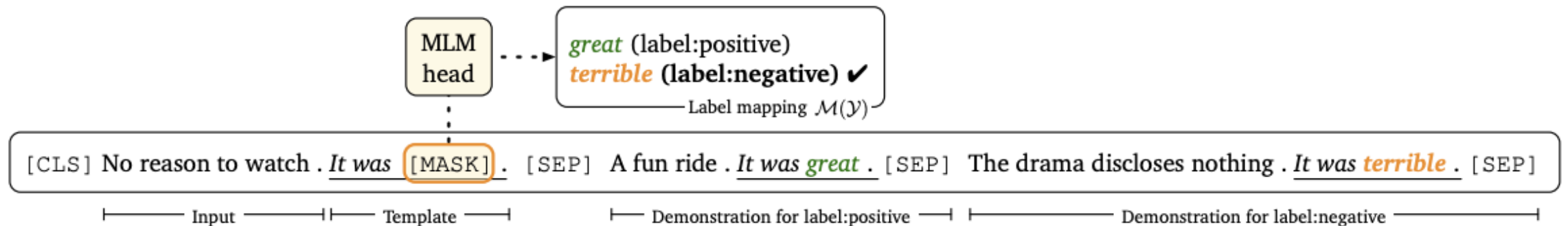
Task 2: In-context learning

- After being provided with a prompt, the model can perform sentiment analysis without the need for training



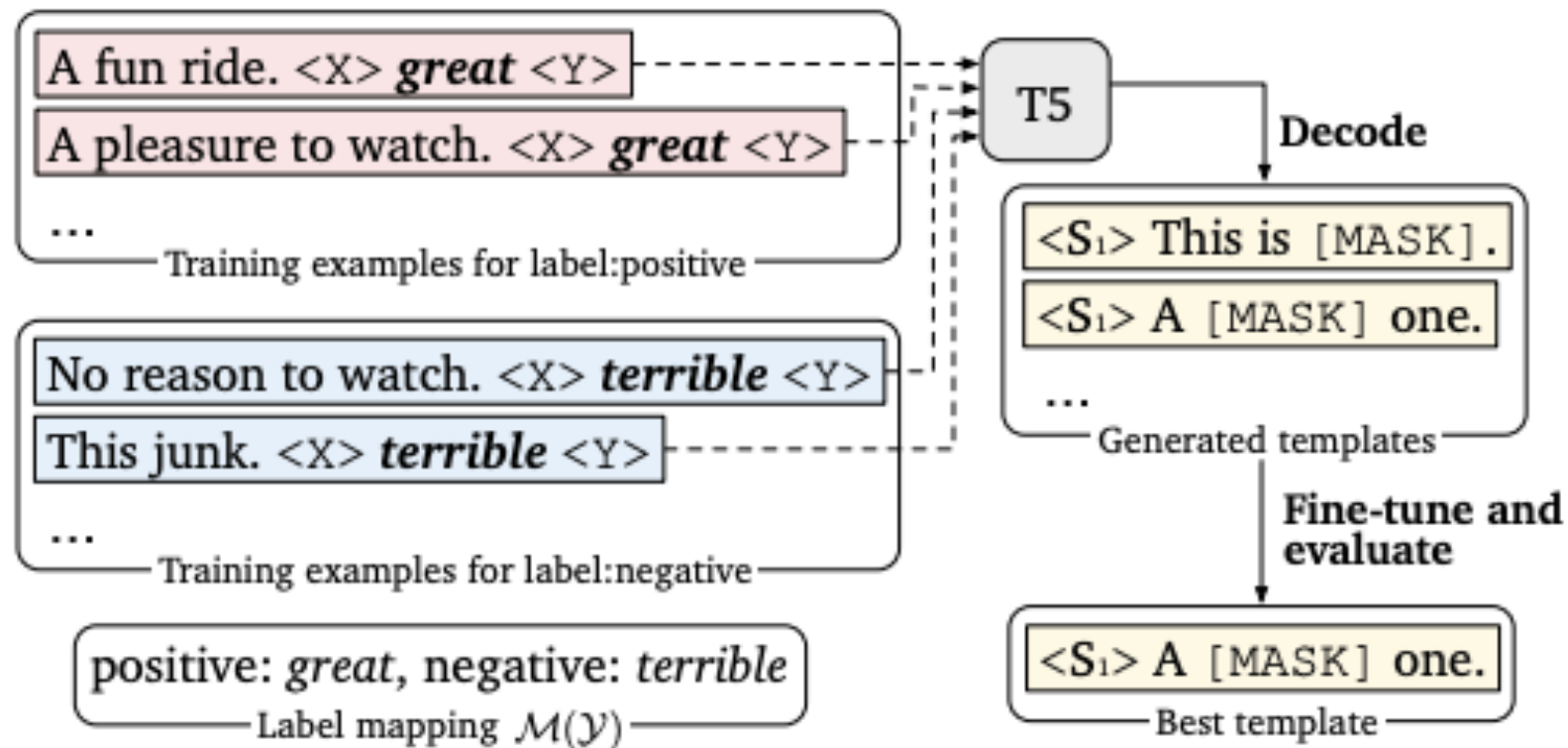
Task 3: LM-BFF

- Utilize prompts and demonstrations for few-shot learning



Task 3: LM-BFF

- Utilize the model to generate the optimal template and verbalizer automatically



Things you cannot do

- You cannot submit results predicted by others.
- You cannot copy trained models from others.
- You cannot copy code from others, internet, GitHub ...
- You cannot collect more texts to train your model in order to boost performance.

Any violation will result in 0 score!

Important issue

- You are not allow to use the model like GPT family and model pre-trained on SST-2 and twitter dataset!!
- You can only use BERT or RoBERTa encoder model !!

Any violation will result in 0 score!

Submission

- Submit your predictions on the test tweets to Kaggle for evaluation.
- Kaggle competition

Task1: <https://www.kaggle.com/t/f072e95f51bc48978225941dba218241>

Task3: <https://www.kaggle.com/t/5b8876ed26fd495b8353ad7ce94b6f65>

- Remember to your SID as the Team Name
- Evaluation metric: accuracy
- Submit your code and report to the CU.
- File name: A6.ipynb, report.pdf

Report

- In Task1, compare the **two** models you employed and provide a brief discussion of your implementation.
- In Task2, you need to try at least **three** different templates and verbalizers to compare how your prompts work with the model. Report your performance in zero-shot, one-shot, and few-shot scenarios, with examples drawn from the training set.
- In Task 3, try at least three different manually crafted templates to compare them with auto-generated templates. Evaluate the performance with different numbers of demonstrations and plot the graph from Figure 3 in the paper (<https://arxiv.org/pdf/2012.15723.pdf>). Also, report your best template and verbalizer.

Grading

- Task1: 40 points (necessary)
- Task2: 32 points (optional)
- Task3: 45 points (optional)
- Report: 15 points
- Bonus points to top 3 teams
- Top 3 teams will share their approaches in class