Writing Assistance Test Assignment - Report

Roman Kovalev

March 27, 2025

1 Introduction

This report outlines the approach and ideas for formality detection. It gives a brief overview of the data used for the assignment and the evaluation measures.

The code with instructions on reproducing the results locally can be found here.

2 Data

For the purposes of this assignment I used the formality scores dataset from the 2016 paper "An Empirical Analysis of Formality in Online Communication" (Pavlick and Tetreault, 2016). It contains 11274 online texts with a formality score from -3 to 3 assigned by experts, where -3 indicates a very informal text and 3 signifies a very formal text.

During the experiment I used the test set comprising 2000 online texts. I also created an additional binary *'formal'* feature with 0 assigned to texts with a formality score lower than 0.0, and 1 - for texts with a score between 0.0 and 3.0.

3 Methods

I implemented three methods of formality detection:

- Flesch Reading Ease Test - a simple formula relying on the total number of words, sentences and syllables in a text. A higher score indicates an easier text to read. While there is no lower limit on how negative the score can be, the highest possible score is 121.22.
- XLM-RoBERTa-based classifier - a binary classifier, trained on XFORMAL - a multilingual formality classification dataset. The classifier returns a dictionary with "formal" and "informal" keys, with probability values for both keys adding up to 1. I treated the prediction as "formal" if the value for "formal" was more than 0.5.
- Binary Classification with Llama-3.2-3B-Instruct. The model is prompted to return "YES" for formal texts and "NO" for informal ones. If the model fails to return one of the options, the example is skipped. 8 examples of formal and informal texts from the train set are also included in the prompt.

4 Results

The sentences with true values and predictions are saved in a .csv file after each evaluation.

Since I had to deal with continuous values, I used Mean Absolute Error and Mean Squared Error scores from the scikit-learn library. The Flesch Reading Ease scores on 2000 datapoints are presented in Table 1.

The Flesch test seemingly does not produce accurate evaluations since the range of values is only 6 between 3.0 and -3.0.

Approach	**Mean Absolute Error**	**Mean Squared Error**
Flesch Reading Ease Test	1.082	1.902

Table 1: MAE and MSE for Flesch Reading Ease Test.

For XLM-RoBERTa-based classifier and Llama 3.2 model I used accuracy, precision, recall and F1 scores, specifically designed for binary classification tasks. The results were obtained based on predictions and the manually created *'formal'* feature from the dataset. The results for all of these metrics are in Table 2.

Metric	**XLM-Roberta-based**	**Llama 3.2**
Accuracy	0.62	**0.67**
Precision	0.55	**0.84**
Recall	**0.93**	0.42
F1-score	**0.69**	0.56

Table 2: Accuracy, Precision, Recall and F1 scores for the XLM-RoBERTa-based classifier and the Llama 3.2 model.

5 Discussion

Due to the time recommendations, I decided to implement one approach from each "category" of methods, with binary classification approaches showing the best results.

The insufficiency of publicly available data with formality scores is a major challenge for formality detection. One of the possible solutions is synthesizing the data, which could be easily undertaken in a project of a wider time scope.

Prompting-based approaches require careful engineering of prompts. When I asked the Llama model to return "YES" and "NO" for formal/informal texts instead of "1" and "0", the model performance vastly improved against all metrics. We also have to take into account the fact that the model can return something different from "YES" or "NO", and these cases should also be dealt with.

Finally, the lack of easily accessible data not in English also does not leave much room for experimenting with the approaches above in other lower-resource languages.