

# **CSCE 685: Directed Studies Final Report**

Siri Pranitha Namburi  
331007601

## **Project:**

Web application for Intimacy Score Prediction of Multilingual Tweets

## **Introduction:**

Effective communication is essential for building and maintaining relationships in social settings. A key factor, in this case, is the level of intimacy of the conversation, and this determines the nature and quality of the relationship. Language plays an important role in conveying social information of intimacy, it encodes both topics of conversation and subtle social cues such as linguistic hedging and swearing.

The project aimed to develop an application that can predict the level of intimacy in text, using NLP models. The project consisted of two main components, the development of the NLP models and the deployment of the models in a web application. My approach builds on previous work in quantifying intimacy score analysis and uses a variety of NLP models to provide accurate predictions. After analyzing different approaches to the task, the best-performing model is selected and integrated into the web application.

## **Previous work:**

The project is built on the recent work by PedroPei in his paper “Quantifying Intimacy in Language”. This paper uses a BERT model that is pre-trained on a large English corpus to quantify intimacy in English datasets. I want to extend this work one step further by expanding it to multilingual settings on tweet data.

## **Methodology:**

To train the backend machine learning model, I used a tweets dataset that already has annotated intimacy scores in six different languages. I have explored different multilingual language models that can be applied to the above-mentioned use case. Some of them include monolingual BERT, multilingual XLM-R, and XLM-T.

The models are then compared using MSE loss and Pearson correlation metric and the best model is used in the backend of the web application to predict the most accurate intimacy score.

## Dataset:

The Tweets dataset used in this project was part of the SemEval 2023 Task 9 challenge and contained approximately 9000 entries across six different languages - Chinese, English, French, Italian, Portuguese, and Spanish. The dataset was distributed evenly across these languages and consisted of tweets mapped to intimacy scores in the range of 1 to 5. A few instances of the dataset are as follows:

Tweet	Intimacy score	Language
Here is a nice equation: $0+0-0-0+0=0$	1	English
@user @user Enjoy each new day! 😊🇨🇦🐛🐭	1.6	English
@user Always my brother.. always be with u 💚	2.5	English
@user 🤔🤔🤔🤔thank you TakaTina my baby kicked...kķkkķķ was kinda worried the whole morning	3.25	English
Las tareas a última hora salen mejor.	1	Spanish
@user @user Lo último era mame de twitter, cada que alguien decía de un abogado era ese licenciado	1	Spanish
@user I think I fell in love with you	4.8	English

As there was no testing data available, the dataset was split into training, testing, and validation sets. To ensure language distribution remained consistent across all splits and phases, a careful splitting strategy was employed. This was crucial to prevent the model from becoming biased towards or against any particular language's intimacy score and to ensure sufficient training in all languages.

## Performance metrics:

**MSE:** MSE (Mean Squared Error) is a popular metric used to evaluate the accuracy of regression models. It measures the average of the squared differences between predicted and actual values.

**Pearson's r:** Pearson's r (Pearson Correlation Coefficient) is a measure of the strength and direction of the linear relationship between two variables. It ranges from -1 to +1, with -1 indicating a perfect negative correlation, +1 indicating a perfect positive correlation, and 0 indicating no correlation. Pearson's r is commonly used to evaluate the performance of machine learning models that predict continuous variables.

MSE and Pearson's r are popular performance metrics because they provide complementary information about the performance of a regression model. MSE measures the accuracy of the predicted values, while Pearson's r measures the correlation between the predicted and actual values. By using both metrics, I aim to compare different models to gain a more comprehensive understanding of the model's performance.

## Baseline:

For the baseline, I utilized the available model from Pedropei to calculate the intimacy scores for the English language. For calculating it for the French language, I used another translation model called Helsinki, which converted the French text into English, and then used the Pedropei model to calculate the intimacy scores. The results below are the test results on English and French languages only.

Languages	MSE Error (English, French)	Pearson's r (English, French)
English, French	0.1086	0.4796

## Approaches:

I explored the following pre-trained NLP models, which are then fine-tuned for the task of predicting intimacy score of tweets data

### Monolingual BERT:

BERT is a pre-trained transformers model that is capable of generating contextualized word embeddings for the English language. It was trained on a large corpus of English text in an unsupervised way, without human labels, through two main objectives: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM involves masking some words in a sentence and asking the model to predict those masked words, allowing the model to learn a bidirectional representation of the sentence. NSP involves concatenating two masked sentences as input, and the model has to predict if the two sentences are adjacent in the original text or not. By pre-training BERT in this way, it learns an internal representation of the English language that can be used for various downstream natural languages processing tasks, such as sentiment analysis or text classification.

In this project, I utilized monolingual BERT models that were pre-trained for three languages: English, French, and Chinese. However, this method has a limitation, as the availability of pre-trained BERT models is limited to only these three languages. Therefore, it becomes difficult to predict the intimacy scores for other languages using this approach. The following are the testing results on single languages only

Monolingual bert used	Bert model name	MSE Error (tested on same language)	Pearson's r (tested on same language)
English	bert-base-uncased	0.033	0.793
French	bert-base-french-europeana-cased	0.082	0.627
Chinese	bert-base-chinese	0.062	0.737

### Multilingual XLM-R:

Insufficient training data and a lack of pre-trained models for foreign languages are common problems with monolingual language models. Hence a multilingual model is better as it can learn semantic relationships in different languages without the need for an extra translation step. Multilingual models utilize the concept of cross-lingual transfer learning to overcome the problem of insufficient training data in monolingual models. Rather than training in each language independently, there are trained in a variety of source languages. Hence it doesn't require any input parameter to specify the text language or any modification to simulate learning other languages.

XLM-R, a language model from Facebook pre-trained on text from Wikipedia and Common Crawl in 100 languages, is a popular multilingual model. I trained XLM-R on there of the language in input - English, Chinese, and French and predicted the intimacy scores in the remaining languages, aka zero-shot predictions.

Testing results on all languages.

Training languages	MSE loss	Pearson's r
English, French	0.112	0.515
English, French, Chinese	0.064	0.724

### Multilingual XLM-T:

The XLM-T (Cross-Lingual Language Model - Twitter) is a multilingual language model specifically designed for analyzing social media data, particularly Twitter. It is an extension of the XLM-RoBERTa-base model and has been trained on a massive corpus of 198 million tweets from more than 30 languages posted between May 2018 and March 2020. The XLM-T model has been fine-tuned to perform well on sentiment analysis tasks and is capable of encoding and processing tweets with their specific characteristics such as hashtags, emoticons, and creative vocabulary and grammar.

Like XLM-RoBERTa, XLM-T is a zero-shot learning model that doesn't require any input parameters or modifications to learn encoding for other languages. It has been trained in a self-supervised manner and is capable of learning from the raw text data. Testing results on all languages.

Training languages	MSE loss	Pearson's r
English, French	0.066	0.707
English, French, Chinese	0.062	0.743

## Results:

Based on the experiments conducted, I have determined that the XLM-T model is the most effective model for predicting intimacy scores across a range of languages, and achieved good performance metrics when trained with my dataset. Therefore, I have decided to use this model for deployment in the backend of my web application.

## Web Application Design:

The web application was developed using Flask, a lightweight web framework for Python. The front end was built using HTML, CSS, and JavaScript. The user interface consists of a single text box where the user can input a text sample, and a button to submit the input.

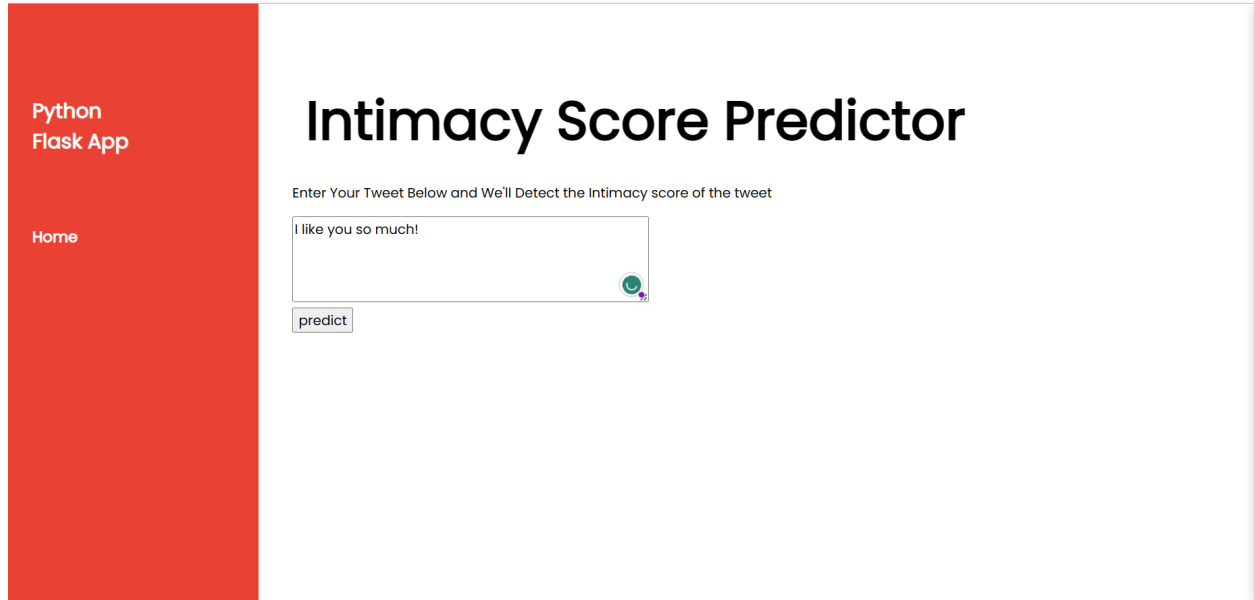
## Application Deployment:

The fine-tuned XLM-T model selected was hosted on the hugging Face model hub, which provides a custom API for making predictions with the model. I integrated the API into the Flask application, which is responsible for collecting user input and sending it to the API for processing. The processed response from the API is then displayed to the user. The Flask application was hosted on PythonAnywhere, a cloud-based platform for hosting Python applications.

## Workflow:

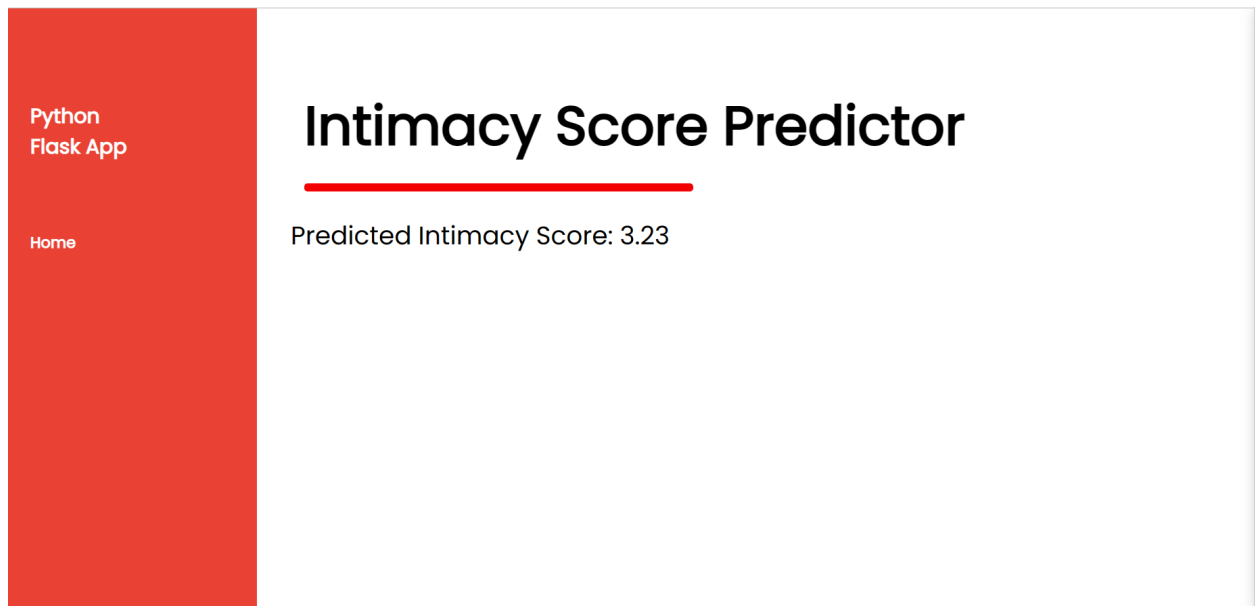
When the user submits a text sample in the web application, the application sends a request to the Hugging Face API with the text as input. The API returns the predicted intimacy score for the text, which is then displayed to the user.

### Inputting the tweet text

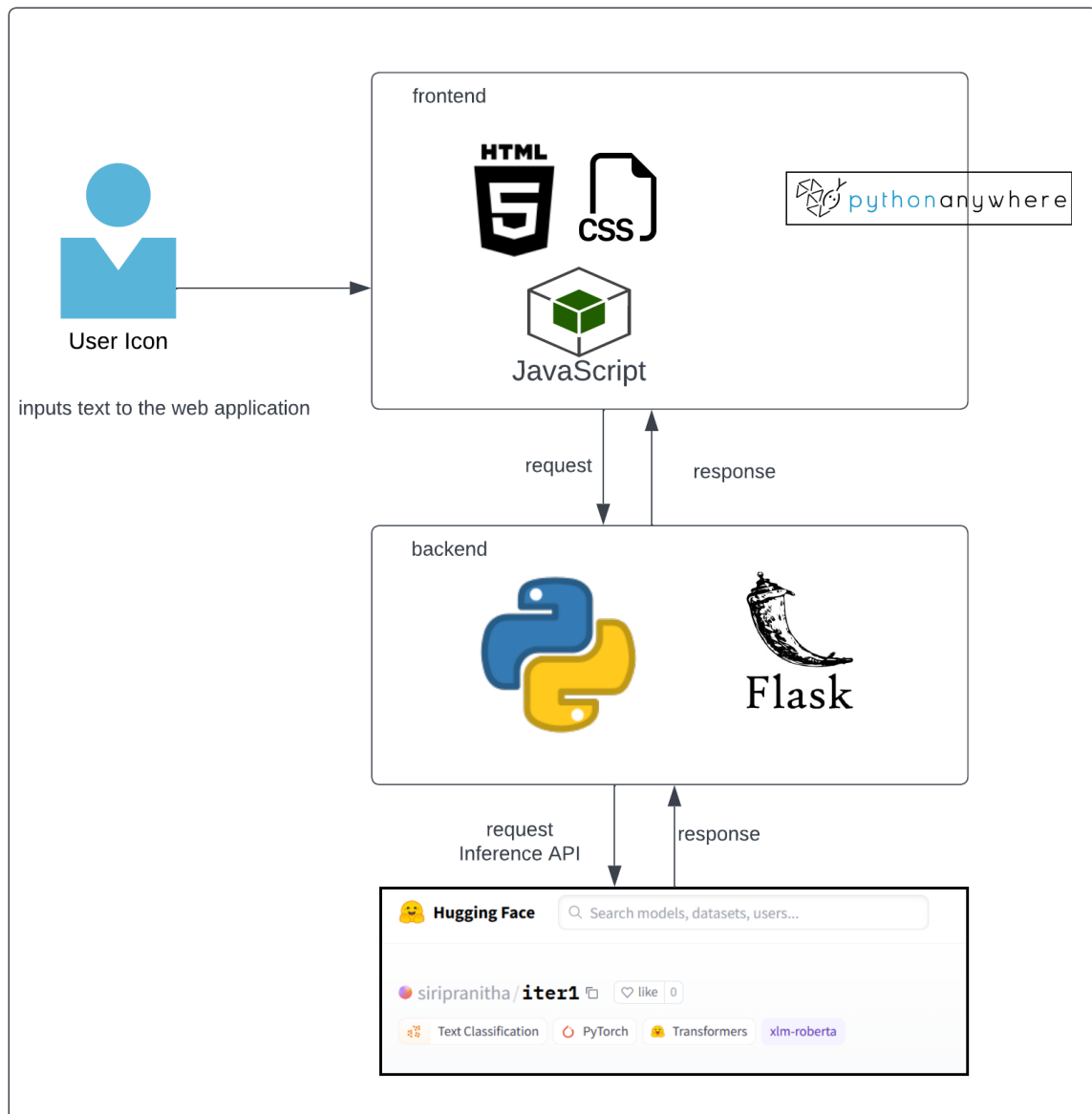


The screenshot shows a web application with a red sidebar on the left containing the text "Python Flask App" and a "Home" link. The main content area has the title "Intimacy Score Predictor" and a subtitle "Enter Your Tweet Below and We'll Detect the Intimacy score of the tweet". Below the subtitle is a text input field containing "I like you so much!" and a "predict" button. A small circular icon with a green checkmark and a purple speech bubble is located to the right of the input field.

The web application returns the predicted intimacy score.



The screenshot shows the same web application as before, but the input field is empty. The main content area now displays "Predicted Intimacy Score: 3.23" below the title "Intimacy Score Predictor". A red horizontal line is positioned above the score.



## Conclusion:

In conclusion, this project demonstrates the use of NLP models in a web application to predict the intimacy score for tweets in multilingual settings. The Multilingual XLM-T model from Hugging Face was found to be effective in predicting intimacy scores in a variety of languages. The web application provides a user-friendly interface for interacting with the model, allowing users to obtain intimacy scores for their text samples quickly and easily.



## Challenges and Future Work:

One of the challenges I faced during the development of the web application was the slow response time of the Hugging Face inference API, which is a free version. As a result, the web application takes longer than desired to provide the intimacy score for a given input text.

Another potential area for future work is the expansion of the NLP model to include more languages. Currently, the model is only trained in English, French, and Chinese text, which limits its usefulness for processing text in other languages. By adding more languages to the training data and fine-tuning the model, I could improve its ability to provide accurate intimacy scores for a wider range of languages.

One more challenge I faced during the project was hosting the web application. While I chose PythonAnywhere as a hosting platform, I encountered some issues during the deployment process. Specifically, difficulties with configuring the application's virtual environment and resolving dependencies. Although I was ultimately able to resolve these issues, they caused some delays in the deployment process and required additional troubleshooting.

## Links:

Github link: [siripranitha/Intimacy-score-detection](https://github.com/siripranitha/Intimacy-score-detection)

Finetuned NLP model: [siripranitha/iter1 · Hugging Face](https://huggingface.co/siripranitha/iter1)

Web application: [Intimacy Score Predictor \(siripranitha.pythonanywhere.com\)](https://siripranitha.pythonanywhere.com/)

Previous work paper: [Quantifying Intimacy in Language - ACL Anthology](#)