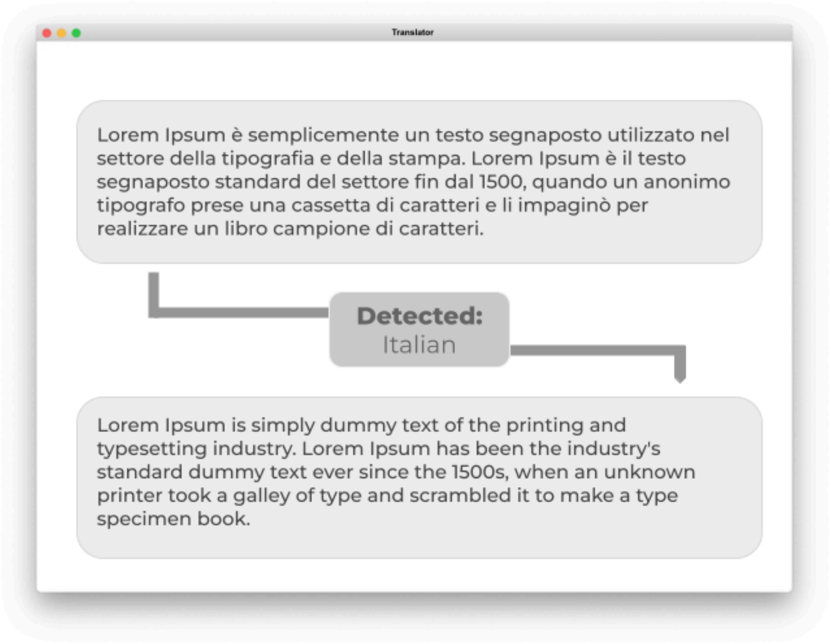
Project Name: Translite
Project Description:



Lorem Ipsum è semplicemente un testo segnaposto utilizzato nel settore della tipografia e della stampa. Lorem Ipsum è il testo segnaposto standard del settore fin dal 1500, quando un anonimo tipografo prese una cassetta di caratteri e li impaginò per realizzare un libro campione di caratteri.

Detected: Italian

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

We aim to build a low requirement, modular, multilingual translation program for small scale applications with low resource availability. Our system will start with a classifier made from an RNN derivative (GRU or LSTM) or an SVM. The choice to use these models will greatly reduce our resource requirements over using a Large Language Model. Our intention is that it will allow us to load specific language models on the fly, specifically into RAM rather than vRAM, allowing it to be used in a wider variety of applications.

The user will input the string of text they wish to translate and the classifier will decipher what language from which the string originates. This will route to the respective translation model and convert the text back to English displaying it to the user. By having a classification network pivot users towards translation networks we further reduce the amount of memory needed at one time, something particularly helpful in applications where RAM is limited. Our intention is to produce a training script to go along with the application, allowing users to modify the language options for their specific use case.

Project Data: For training data we intend to use publicly available English translation language datasets from Kaggle. This will allow for more seamless integration into a training script, enabling users to expand or reduce the available languages much more easily. The expected formats for all of the datasets will be that they require an English column and a translation column. An example of such a dataset would be https://www.kaggle.com/datasets/dhruvildave/en-fr-translation-dataset for English to French translation. We will ensure that these languages will be directly ingestible by a Word2Vec or similar embedding model. For the classification model we will be able to associate each dataset with an integer linked to that specific language. Thus the training set for the classification model will end up being a column of the non-English data-points, and a column with the integer associated with the dataset/language from which they came as our end goal will be to predict that integer based on the user input which will be used to select the relevant translation model.

Proposed Method: Our main plan is to get the core foundation of our pipeline fleshed out and in a user friendly format. This will consist of designing a Colab Notebook dedicated to expediting the training and testing process for all users. The Colab Notebook will be made in a user friendly format and split into three sections, configuration, training, and testing.

The configuration section will allow the user to set and select their languages for their particular application. We will make an effort to abstract a large portion of the dataset processing from the user to make it as simple as possible. The following section will be training, and the plan is to take a similar approach of abstracting it. Once the user has configured the languages and loaded the datasets they will be able to fire off a training script which will first assign an integer to each language, then construct an aggregate dataset of the non-English columns of the individual datasets. This will be fed into the classification model trainer. Once that finishes training, each of the translation models will be trained individually on their respective datasets. They will all be outputted and downloadable with a configuration file.

The final section and arguably the most important part of the process will be the demonstration of the pipeline. For the users this will demonstrate how the parts function together (query -> classifier -> translator -> output). This will allow for testing and optimization for the end user, and allow us to achieve our final goal of having a simple fleshed out proof of concept implementing models trained on our script to show how our users might take the models and code provided to implement their own pipeline.

Related Work:

Khan, A., & Sarfaraz, A. (2019). RNN-LSTM-GRU based language transformation. Soft Computing, 23(24), 13007-13024.

This paper explores the efficacy of using RN derivatives (LSTM/GRU) to provide machine translations, particularly Urdu translations. This concept is the core of our pipeline and their results and findings should prove useful to us.

—

Can, E. F., Ezen-Can, A., & Can, F. (2018). Multilingual sentiment analysis: An RNN-based framework for limited data. *arXiv preprint arXiv:1806.04511*.

This paper explores multilingual sentiment analysis. We hope to use their findings to train a classification model to detect languages, rather than emotions, using a similar premise.

—

Singh, S. P., Kumar, A., Darbari, H., Singh, L., Rastogi, A., & Jain, S. (2017, July). Machine translation using deep learning: An overview. In *2017 international conference on computer, communications and electronics (comptelix)* (pp. 162-167). IEEE.

This is an overview of the fundamentals of using deep learning to translate human language. We hope to use their proposed Recursive Recurrent Neural Networks to power our translation models.

—

Wang, J. H., Liu, T. W., Luo, X., & Wang, L. (2018, October). An LSTM approach to short text sentiment classification with word embeddings. In *Proceedings of the 30th conference on computational linguistics and speech processing (ROCLING 2018)* (pp. 214-223).

Their proposed method of using an LSTM provides an idea of how our pipeline will look. In particular their approach to using a Word2Vec Model to represent their multilingual embeddings will likely help us in planning our embedding process.

—

Ambikairajah, E., Li, H., Wang, L., Yin, B., & Sethu, V. (2011). Language identification: A tutorial. *IEEE Circuits and Systems Magazine*, *11*(2), 82-108.

While being a much older paper, this explored using an SVM to perform language classification, which may prove to be a smarter approach given our goal of minimizing resource usage. This may be another avenue we explore over using an RNN derivative for classification.

Team Roles:
Matthew Hambrecht - Colab Training Script Developer/Technical Writer
Aryan Singh - Proof of Concept Application Developer/Technical Writer
Gabe Clark - Colab Training Script Developer/Technical Writer
Sai Parthasarathy - Proof of Concept Application Developer/Technical Writer