Group 10: Language Translation Service

Swetha Reddy Ganta
sganta3@buffalo.edu

Pranavi Sriya Vajha
pvajha@buffalo.edu

Abstract

Language Translation Service project aims to develop a neural machine translation service capable of translating text from one language to another. Leveraging advanced Natural Language Processing (NLP) techniques and sequence-to-sequence models, we implemented and fine-tuned various models to achieve optimal translation accuracy. Our primary focus was on translating English to Swedish, French, and Bulgarian using publicly available datasets.

1 Dataset

1.1 Description of the Dataset:

We conducted experiments with train, test and validation sets of Europarl dataset, a parallel corpus extracted from the European Parliament proceedings. This dataset is publicly available and provides aligned sentences in multiple languages, making it ideal for training machine translation models.We used English to French, English to Sweden and English to bulgarian datasets for our project.

1.2 Data Engineering Used for the Dataset:

The data engineering process for this dataset involves several key steps:

1. **Data Collection:** We utilized the Europarl dataset, which consists of parallel corpora from European Parliament proceedings, providing aligned sentences in multiple languages.
2. **Data Cleaning:** The data was cleaned by removing all the unnceccesary columns and by keeping the language pair columns, later, renamed the columns for better readability. Then we have removed columns with null values and removed the rows which does not contain any alphabetical characters.
3. **Data Preprocessing:**
   - Tokenization: Sentences were converted into a sequence of tokens, which are essential for processing by machine learning models.
   - Padding: Implemented padding to ensure that all input sentences had uniform lengths, facilitating batch processing.
4. **Handling Missing Data:** Mostly the data is clean and contains only small proportions of data which are missing.So, we removed the rows which contains null values.

2 Model Description

In this section, we will review relevant AI algorithms, that were used in our project.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

2.1 Model Architecture

We experimented with various sequence-to-sequence models, including those with attention mechanisms which are used in language translation. Among these models, we focused on the MarianMT architecture because of its proven efficiency in translation applications. It uses encoder-decoder network. The encoder processes the input sentence and understand the syntactic and semantic relations. The architecture contains 6 encoder and 6 decoder layers. The decoder takes the encoded representations of the encoder and generates output.

2.2 List of models used

1. **Helsinki-NLP/opus-mt-en-sv:** This is a powerful tool for translating English into Swedish, built on the advanced Transformer architecture. It's trained on a wide range of text sources, making it versatile for everything from everyday communication to professional translations.
2. **Oskarandrsson/mt-en-sv:** For English to Swedish translation
3. **Helsinki-NLP/opus-mt-en-fr:** For English to French translation.
4. **Skier8402/marian-kde4-en-to-fr:** Fine-tuned model for English to French translation.
5. **Helsinki-NLP/opus-mt-en-bg:** For English to Bulgarian translation.

2.3 Best Model Selection

The choice of the best model was based on the validation loss and BLEU score. For English to French translation, the Skier8402 model showed the best performance with a validation loss of 0.2063 and a BLEU score of 0.2076. While in English to Swedish, Helsinki model showed better performance with a BLEU score of 0.0229, but the validation loss was least for Oskarandrsson/mt-en-sv model.

2.4 Algorithm Description

The Helsinki-NLP/opus-mt models are the state-of-the-art neural machine translation algorithm developed by Helsinki-NLP for translating . They utilize the Transformer architecture with attention mechanisms to focus on relevant parts of the input text, enhancing translation accuracy. Trained on the extensive OPUS dataset, it captures the nuances of both languages, achieving competitive BLEU scores. The model is available on the Hugging Face Model Hub, allowing easy access and can be fine-tuned for specific applications. This makes it a best choice for language translation applications .

3 Loss Functions

3.1 Chosen Loss Function

We primarily used the cross-entropy loss function, which is well-suited for sequence to sequnece tasks like language translation. This loss function measures the difference between the predicted probability distribution and the actual distribution.

The cross-entropy loss function for sequence-to-sequence models can be expressed as:

$$L = -\frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{V} y_{t,j} \log(\hat{y}_{t,j})$$

where:

- T is the length of the sequence (number of time steps).
- V is the size of the vocabulary (number of possible tokens).
- $y_{t,j}$ is an indicator variable that is 1 if token j is the true token at time step t, and 0 otherwise.
- $\hat{y}_{t,j}$ is the predicted probability of token j at time step t.

3.2 Other Loss Functions Tried

We experimented with mean squared error (MSE) but found it less effective for our task compared to cross-entropy loss.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{1}$$

where n is the number of data points, y_i is the actual value, and $\hat{y}_i$ is the predicted value.

3.3 Innovation on Loss Function

We implemented label smoothing to improve generalization. Label smoothing replaces the hard 0 and 1 targets with values like 0.9 and 0.1, which helps the model to not become overconfident in its predictions.

4 Optimization Algorithm

4.1 Chosen Optimization Algorithm

We chose Adam (Adaptive Moment Estimation) as our primary optimization algorithm due to its effectiveness with large datasets and fast convergence. Adam combines the best aspects of AdaGrad and RMSProp, adapting the learning rate for each parameter dynamically, which facilitates quicker optimization and addresses the issue of vanishing learning rates efficiently.

4.2 Other Optimization Algorithms Tried

1. **SGD (Stochastic Gradient Descent):** A basic optimization method that updates model parameters using small, random samples of data to gradually minimize the loss function.
2. **RMSprop:** Attempted to address some issues with adaptive learning rates but did not outperform Adam.

4.3 Innovation on Optimization Algorithm

We implemented learning rate schedules to dynamically adjust the learning rate during training. This helped in faster convergence and avoiding local minima.

5 Metrics and Experimentation Results

5.1 Metrics used

We used the BLEU (Bilingual Evaluation Understudy) score to measure the quality of translations. The BLEU score compares the n-grams of the predicted translation with those of a reference translation.

5.2 Experimental Results

The following tables illustrate the results of the three language language translations based on their validation loss and BLEU scores.

Model	Validation loss	BLEU score
Helsinki	3.0053	**0.0229**
Helsinki(optimizer=SGD, lr=1e-6)	2.7019	0.0222
Oskarandrsson/mt-en-sv	**2.6224**	0.0225

Table 1: English to Swedish

- For English to Swedish, the Oskarandrsson/mt-en-sv/mt-en-sv model stands out with the lowest validation loss of 2.6224 and a BLEU score of 0.0225, making it the best choice. The Helsinki model with hyperparameter tuning also performs well with a validation loss of 2.7019 and a BLEU score of 0.0222.

Model	Validation loss	BLEU score
Helsinki	3.1017	0.0189
Helsinki(optimizer=SGD, lr=1e-6)	3.1390	0.0187
Marian	**2.0636**	**0.0207**

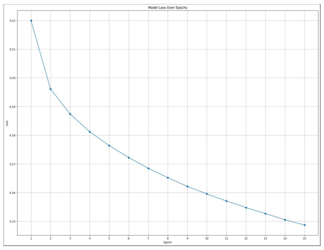
Table 2: English to French

Model	Validation loss	BLEU score
Helsinki	5.89687	0.0121

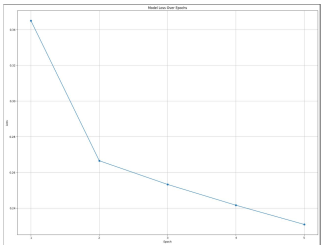
Table 3: English to Bulgarian

- For English to French, the Marian model is the top performer, achieving the lowest validation loss of 2.0636 and the highest BLEU score of 0.0207, indicating its superior translation quality.
- For English to Bulgarian, only the Helsinki model is experimented, with a validation loss of 5.89687 and a BLEU score of 0.0121.

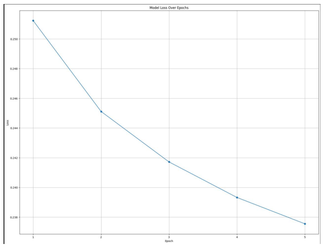
The following graphs are plotted for Model loss against epochs:



(a) English-Bulgarian



(b) English-French



(c) English-Sweden

These graphs highlight the effectiveness of the training process, with each model showing a decrease in loss, which is a key indicator of learning and performance enhancement.

6 Webpage

We have deployed a web service by incorporating our results into it. The web page asks for an English statement, which needs to be translated and then translates it to 3 different languages,namely, Bulgarian, Swedish and French (Figure 2).

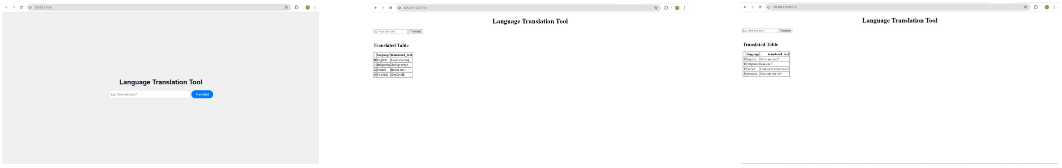
7 Github

Here is the Github link to our project: Group 10: Language Translation Service

8 Conclusion

In this project, we developed a neural machine translation service capable of translating text from English to Swedish, French, and Bulgarian. We used the Europarl dataset, which provided a rich source of parallel text, essential for training our models. Through proper data preprocessing and engineering, we ensured the dataset was in optimal condition for training.

We explored several sequence-to-sequence models with attention mechanisms, focusing on their ability to handle the nuances of different languages. The encoder-decoder architecture, enhanced by attention mechanisms, proved effective in generating translations. We experimented with various models, including the Helsinki-NLP/opus-mt-en-sv, Oskarandrsson/mt-en-sv/mt-en-sv, and Marian-en-to-fr, among others. Each model brought unique strengths, with the Marian model standing out in English to French translation due to its fine-tuning on the KDE4 dataset.



Language Translation Tool

(a) Input

Language Translation Tool

Translated Table

	language	translated_text
0	English	Good evening
1	Bulgarian	Добър вечер.
2	French	Bonne nuit
3	Swedish	God kväll.

(b) Sample output 1

Language Translation Tool

Translated Table

	language	translated_text
0	English	How are you?
1	Bulgarian	Как сте?
2	French	Comment allez-vous?
3	Swedish	Hur står det till?

(c) Sample output 2

Figure 2: Comparison of input and sample outputs

We chose the cross-entropy loss function and used techniques like label smoothing to help our models generalize better and avoid overfitting. The Adam optimizer was our primary choice due to its adaptive learning rate properties, although we also experimented with SGD and RMSprop.

The experimental results revealed distinct patterns. For English to Swedish, the Oskarandrsson/mt-en-sv/mt-en-sv model showed slight improvements over the Helsinki models, though all models demonstrated modest BLEU scores. The English to French translation was notably successful with the Marian model, achieving the highest BLEU score among all models tested.

The insights gained from this project provide a solid foundation for future advancements in machine translation, paving the way for more accurate and reliable language translation services.

9 References

- Pytorch
- Helsinki-en-bg model
- Helsinki en-fr model
- Helsinki en-sv model
- Marian-en-fr model