Semantic Role Labeling Homework 2 - NLP 2022

Mirza Mujtaba Hussain
Email: mirza.1989740@studenti.uniroma1.it

1. Introduction

Semantic Role Labeling (SRL) is a task in natural language processing (NLP) that aims to identify the underlying semantic relationships between the arguments and the predicate in a sentence.A predicate is a verb or noun that expresses an action, occurrence, or state-of-being while as arguments are the nouns, pronouns, or phrases that provide information about the predicate.This task plays an important role in understanding the meaning of a sentence, and is crucial for several NLP applications.

SRL is usually divided into four subtasks: Predicate identification: This involves identifying the predicate in the sentence and its corresponding arguments. Predicate disambiguation: This involves determining the correct meaning of the predicate in the context of the sentence. Argument identification: This involves identifying the arguments associated with the predicate in the sentence. Argument classification: This involves classifying the arguments into specific semantic roles such as agent, patient, instrument, etc.

In this project we will perform Predicate disambiguation,Argument identification and Argument classification on an English dataset. Later in this project we will also explore that how we can use the trained model in English language and fine tune it on a secondary language with less samples, in our case it is French and Spanish

2. Data Encoding

For this project we have 3 datasets in 3 different languages, where dataset in english is our primary dataset, whereas we have parallel corpus in French and Spanish Language.

In this project,since I am dealing with multiple languages I used the XLM-RoBERTa (Cross-lingual Representations from Transformers) to encode the input text and generate the word embeddings . It is a transformer-based language model pre-trained on a large multilingual corpus which makes it ideal for this project. XLM-RoBERTa is a variant of BERT and the main reason behind using it is to get contextualized word representation.

As mentioned, It is a variant a BERT, as as we do when using BERT, we got the embeddings by summing the outputs from last 4 layers .In our project. This approach allowed us to capture the most relevant and contextual information from the input text, which is crucial for our task.

3. Training

In the training process we train the models on primary dataset. We have divided the project in two tasks. In task 1, Given predicate location and predicate sense, we identify and classify the arguments.While in task 2, Given just the predicate location, we do predicate disambiguation, argument classification and argument identification:

- **Task 1:** As mentioned ,here given predicate location and predicate sense, we identify and classify the arguments. While the training process , after getting the embeddings from transformer network whose weights are frozen, we feed it into a Bi-LSTM layers.The inputs given to this layers are the outputs from xlm-RoBERTa , part of speech tag of each token and predicate location and sense. The outputs from Bi-LSTM are fed into two MLP layers and then we get output for each token.The model here is trained simuntaneosly to identify argument location and its class.

- **Task 2:** The method of this task is really similar to task 1 , but here the inputs given are only predicate location, part of speech tags obtained from the Spacy libraray and location of predicate.In this task , the model is simuntaneosly trained to identify the predicate sense, argument location and its class.

While training the Adam optimizer was utilized to update the model parameters. Dropout was also employed as a regularization technique to prevent overfitting. Also the training data was split into training and validation part.

4. Fine tune

Now for each task, we have a model trained on primary dataset ,English, we finetune it for secondary languages(spanish and french). For fine-tuning we used the Multi-task learning approach and used two strategies:

- **Strategy 1:** Using this strategy, we used the trained model as our base model, removed its last layer and added a new layer. We also experimented with freezing the weights of the base model.After this we trained the model on the secondary language.

- **Strategy 2:** In this strategy, we used the base model as it is and also experimented with freezing the weights of all layers except the last one.

5. Results

During this project, I used scikit-learn to calculate the f1-score on a separate test set. In task 1 , the strategy two without freezing the parameters of base model produced the best results for the secondary languages. While as for task 2, the strategy one was more useful for getting slightly better results. Although there was no siginificant difference found when experimenting with freezing the layers of base model.

Language	Englsih	Spanish	French
Argument Identification	0.90	0.81	0.80
Argument Classification	0.82	0.70	0.69

The table above displays the f1 score calculated for each subtask on each language calculated while doing task 1.The secondary languages are fine tuned using strategy 1

Language	Englsih	Spanish	French
Argument Identification	0.90	0.84	0.82
Argument Classification	0.82	0.71	0.69

The table above displays the f1 score calculated for each subtask on each language calculated while doing task 1.The secondary languages are fine tuned using strategy 2

Language	English	Spanish	French
Predicate disambiguation	0.85	0.57	0.54
Argument Identification	0.90	0.76	0.76
Argument Classification	0.81	0.62	0.58

The table above displays the f1 score calculated for each subtask on each language calculated while doing task 2.The secondary languages are fine tuned using strategy 1

Language	English	Spanish	French
Predicate disambiguation	0.85	0.44	0.50
Argument Identification	0.90	0.65	0.71
Argument Classification	0.81	0.57	0.51

The table above displays the f1 score calculated for each subtask on each language calculated while doing task 2.The secondary languages are fine tuned using strategy 2

6. Conclusion

In this project, we explored the use of xlm-RoBERTa embeddings and used it for different languages. We also explored on we can transfer the knowledge acquired by the model on one language to another language with less annotated data.The results in task 2 were surprisingly comparable in secondary languages with that of task 1, even though we were not using predicate sense information while training. Whereas in case of primary language, the results are similar. We also found out how one fine tuning strategy worked better than another in different scenario. For task 1, the strategy 2 performed well but for task 2 , the fine tuning strategy 1 performed better results