Documentation

Language Translation using Transformers

Jupyter Notebooks contents:[Source code]

The source code has been divided into different folders grouped by language for ease of interpretability and is uploaded as separate jupyter notebook files.

English-Dutch → 'English-Dutch-opus-t5.ipynb' – English to Dutch translation using opus book dataset and t5 transformer.

→ 'English-Dutch-opus-marianMT.ipynb' - English to Dutch translation using opus book dataset and marianMT transformer.

English-French → 'English-French-kde4-t5.ipynb' – English to French translation using kde4 dataset and t5 transformer.

→ 'English-French-opus-marianMT.ipynb' - English to French translation using kde4 dataset and marianMT transformer.

English-German → 'English-German-opus-marianMT.ipynb' - English to German translation using opus book dataset and marianMT transformer.

→ 'English-German-opus-t5.ipynb' - English to German translation using opus book dataset and t5 transformer.

→ 'English-German-wmt16-marianMT.ipynb' - English to German translation using wmt16 dataset and marianMT transformer.

→ 'English-German-wmt16-t5.ipynb' - English to German translation using wmt16 dataset and t5 transformer.

Table representing dataset libraries used for respective languages:

	opus_books	wmt16	kde4
English-Dutch	✓		
English-French			✓
English-German	✓	✓	

The Dataset:

All the datasets used in the source code have been directly loaded into notebooks from the 'hugging face' website using the 'load_dataset' feature of the 'datasets' library. Direct links for the same have been provided in the 'Dataset.md' file of the repository. Please refer to data cards of the respective datasets for more information on the format and contents of the data files.

Note: The 'hugging face' interface also provides provision for applying filters when searching for datasets and pre-trained models with respective to required NLP task.

Some Common Functions defined in all the notebooks:

'preprocess' : Used to perform pre processing on all instances of the dataset by tokenizing both input and target sequences.

'compute_metrics' : The default 'compute_metrics' function of the model 'train' feature returns the training and validation loss by default. In this case the function has been modified to return metrics like bleu score and meteor score.

Checkpoints:

- ✓ Make sure all necessary libraries are installed before running the notebooks, if not run the following commands from the jupyter notebook interface.

```
!pip install torch
!pip install transformers
!pip install datasets
!pip install numpy
```

- ✓ The input model variable for both the `AutoTokenizer.from_pretrained()` and the `AutoModelForSeq2SeqLM.from_pretrained()` should belong to the same model, as each model is provided with it's own tokenizer customized for it.

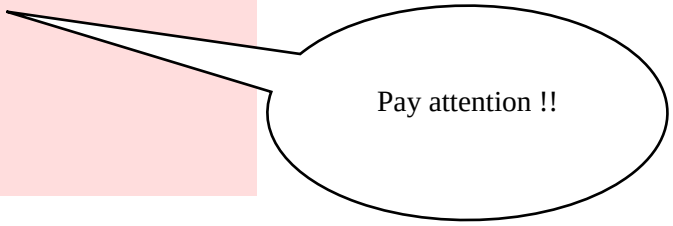
Result:

The metric scores like training loss, validation loss, bleu score , meteor score for each of the models have been displayed at the end of the respective notebooks and will also be explained in detail as part of the Project report.

Troubleshooting:

The following was a common error observed while running ‘compute_metrics’ function part of the notebooks.

```
[nltk_data] Error loading wordnet: Remote end closed connection
[nltk_data]     without response
[nltk_data] Downloading package punkt to C:\Users\Sharanya
[nltk_data]     Manohar\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package omw-1.4 to C:\Users\Sharanya
[nltk_data]     Manohar\AppData\Roaming\nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
```



On observing the following error , please re-run the codeblock. It is a common error observed when there is an interruption of connection from the ‘wordnet’ source side.