AI Machine Learning & Analytics
Jesús López Cubas
Report on Natural Language Processing Assignment

This report summarizes my end-to-end pipeline for classifying the primary emotion expressed in tweets, the key experimental results, and the performance of my final model visualized through training curves.

Initially, the modeling and data preparation processes were carried out in a precise sequential order. I used a text-cleaning function to lowercase all characters, remove URLs and non-alphabetic symbols, and collapse extra whitespace after loading the supplied CSV files. To lessen noise from outlier lengths, tweets with fewer than three tokens or more than sixty tokens were eliminated. With indices 0 and 1 set aside for padding and out-of-vocabulary tokens, respectively, a vocabulary of the 10,000 most common tokens was created. After that, each tweet was mapped to a fixed-length sequence of 200 token indices, with longer sequences being truncated and shorter ones being padded. LabelEncoder was used to encode emotion labels into integer targets. The dataset was split into training and validation sets in an 80/20 stratified manner (random seed 42) to preserve class proportions.

Then, using a linear output layer across six emotion classes, a 256 unit hidden state, a 300 dimensional embedding layer, and 50% dropout, I created a bidirectional GRU classifier. Per class weights were calculated and applied to the cross entropy loss to rectify the class imbalance. When validation macro F1 plateaued, I employed a ReduceLROnPlateau scheduler (factor 0.5, patience 2) to lower the learning rate. The model was trained using the Adam optimizer at a learning rate of 4e 4. After four consecutive epochs without improvement, early stopping was applied for a total of 30 epochs.

The following table displays the most significant hyperparameter trials, including the final configuration highlighted in bold:

Trial	Embedding Dim	Hidden Dim	Learning Rate	Batch Size	Val Macro-F1
1	100	128	1e-3	32	0.865
2	200	256	5e-4	64	0.889
3	**300**	**256**	**4e-4**	**64**	**0.897**

Lastly, Figure 1 displays the validation macro F1 progression for the chosen model along with the training and validation loss curves. After the scheduler lowered the learning rate, the model continued to fine-tune the classification boundaries, and loss convergence was achieved by epoch 12. The checkpoint that produced a validation macro F1 of 0.897 and a balance between underfitting and overfitting was the result of early stopping at epoch 16.

```
Install:

pip install
git+https://github_pat_11BGFAT6Y0JR2JpP7TlO9U_eeY2s5wkduxF2l5nAEFpaFlO
vefpnbAZcl45HJ2Uacf5MURI7YVtWbIhDf8@github.com/JesusLopezCubas/KAGGLEC
OMPETITION.git
```