THE AMERICAN UNIVERSITY IN CAIRO
الجامعة الأمريكية بالقاهرة

TOXIC COMMENTS CLASSIFIER

Mohamed Hashish, Sherif Wessa

PROBLEM

Toxic comments encompass a wide range of harmful content, including abusive language, insults, and offensive remarks meant to harass or threaten others.

IMPACT

This issue has led to the discouragement of many individuals from openly sharing their ideas and opinions due to concerns about receiving hate comments, which, in turn, negatively impacts their mental and emotional well-being.

CHALLENGE

The primary issue we confront is the effective classification and detection of toxic comments on social media and online platforms.

DATASETS

We are using the Jigsaw toxic data as our Dataset that has a large number of Wikipedia comments which have been labeled by human raters for toxic behavior. It is labeled by 6 types of toxicity which are: toxic, severe_toxic, obscene, threat, insult, identity_hate.

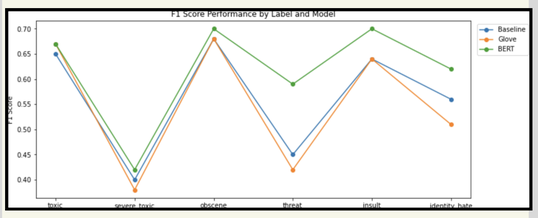
STATE OF THE ART

	$\overline{\text{F1}}$	F1_{toxic}	$\text{F1}_{\text{engaging}}$	F1_{fact}
200 GELECTRA multi-label	0.717	0.713	0.690	0.748
200+200 GELECTRA/GBERT multi-label	**0.726**	0.716	**0.699**	**0.763**
30+30 GELECTRA/GBERT single-label	0.699	**0.718**	0.658	0.723
corrected scores	**0.727**	**0.717**	0.697	**0.768**

ORIGINAL MODEL

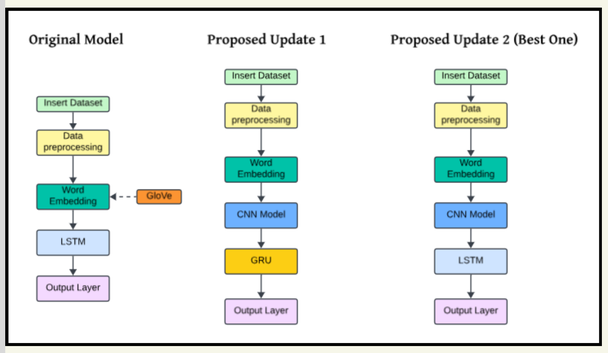
Three different models using TensorFlow to address the challenge:

- MODEL I: a baseline approach that utilized a Bidirectional Long Short-Term Memory (LSTM) network with embeddings trained from scratch.
- MODEL II: a variation of the baseline approach that incorporated Glove's pre-trained embeddings with the Bidirectional LSTM architecture.
- MODEL III: the well-known BERT model, which is capable of producing state-of-the-art results on a range of NLP tasks including text classification.

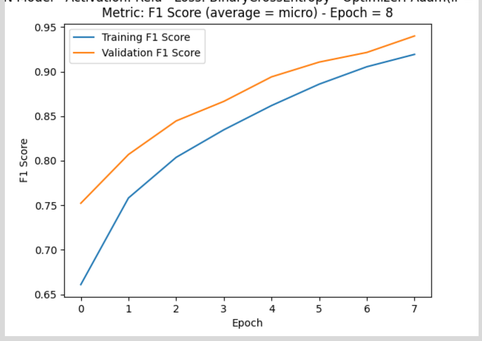


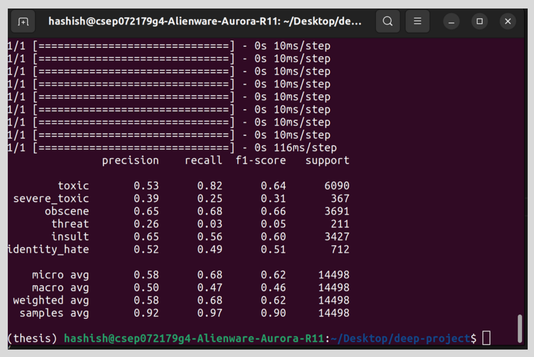
UPDATED MODEL

The original baseline model used a Bidirectional LSTM model with GloVe word embeddings. The first proposed update included an GRU model combined with a CNN model without GloVe word embeddings, and then we introduced a new update which is combining an LSTM model with a CNN model which turned out to be of better accuracy and results.



RESULTS





```
hashish@csep072179g4-Alienware-Aurora-R11: ~/Desktop/de...
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 10ms/step
1/1 [==============================] - 0s 116ms/step
              precision    recall  f1-score   support

       toxic       0.53      0.82      0.64      6090
severe_toxic       0.39      0.25      0.31       367
     obscene       0.65      0.68      0.66      3691
      threat       0.26      0.03      0.05       211
      insult       0.65      0.56      0.60      3427
identity_hate       0.52      0.49      0.51       712

   micro avg       0.58      0.68      0.62     14498
   macro avg       0.50      0.47      0.46     14498
weighted avg       0.58      0.68      0.62     14498
 samples avg       0.92      0.97      0.90     14498

(thesis) hashish@csep072179g4-Alienware-Aurora-R11:~/Desktop/deep-project$
```

CONCLUSION

After trying different architectures and hyperparameters, we found out that the hyperparameters that were used in the original baseline model weren't the best for our problem statement. Upon changing the hyperparameters, we found out that the highest validation accuracy was acheived by the LSTM (0.96), and the second highest validation accuracy was the CNN model that we introduced (0.94). What we learned from these experiments is that it was our first time to combine different architectures together, and we also learned how to use virtual environments to run our code on the GPU.

FUTURE WORK

For future work, we would like to try different architectures and hyperparameters to see if we can get a better validation accuracy, and we want to have a chance to change the hyperparameters of the BERT model since we were limited by time as it required half an hour for running a single epoch.

REFERENCES

OBaseline Model: https://github.com/alessiococchieri/toxic-comment-classification/tree/main
Dataset: https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data
SOTA: https://aclanthology.org/2021.germeval-1.16.pdf