Identifying Russian Trolls on Reddit with Deep Learning and BERT Word Embeddings

Henry Weller, Jeff Woo
CS224N: Natural Language Processing With Deep Learning, Stanford University 2019

Motivation & Goals

- **Goals:**
  - Use deep learning to identify Russian Trolls based on their comments
  - Use BERT[1] to bypass significant data limitations
- **Motivation:**
  - Russian trolls are a major threat to social media platforms and remain at large on Reddit
  - Troll detection on Reddit is basically nonexistent!

Dataset

- ~7000 comments from over 900 banned Russian Trolls on Reddit from the 2017 Transparency Report[2]
- ~7000 randomly sampled human comments from Reddit[3]
- 64% Train, 16% Dev, 20% Test

Models

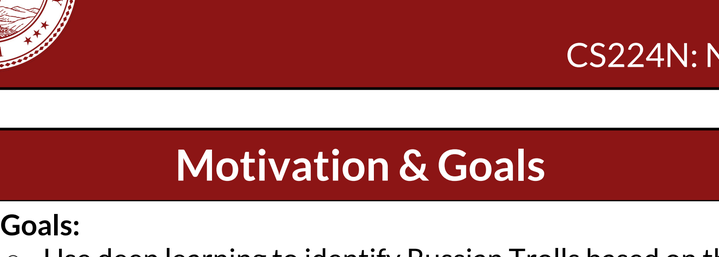


Figure 1: High level overview of model architecture

- BERT layer to find comment word embeddings
- Middle Layer Examples: RNN, CNN, RCNN[4]
  - RCNN: bidirectional LSTM + Max pooling
- Dropout and early stopping for regularizations
- Binary linear classifier layer

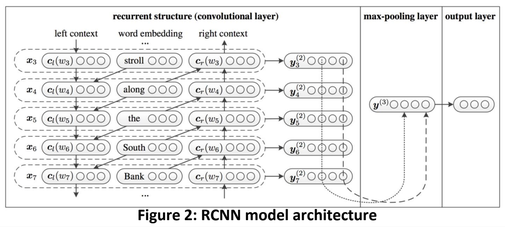
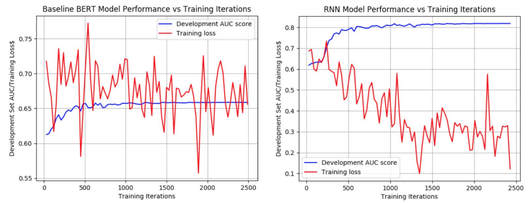


Figure 2: RCNN model architecture

Results

	AUC
Optimized Forest Classifier Baseline	0.74
BERT Baseline	0.659
BERT + LSTM	0.805
Vanilla BERT + CNN	0.826
Optimized BERT + CNN	0.843
Vanilla BERT + RCNN	0.836
Optimized BERT + RCNN	**0.846**



(a) BERT + Classifier (b) BERT + LSTM + Classifier

(c) BERT + CNN + Classifier (d) BERT + RCNN + Classifier

Figure 3: Training Loss Curves over time + AUC scores

Optimization

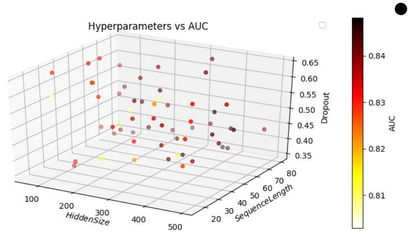


Figure 4: Random Search Hyperparameter Optimization Visualized

- Out of the 60 models we trained during Randomized Hyperparameter Search, we found a dropout rate of 0.445, comment size of 70 words, and hidden size of 415 to be optimal hyperparameters.

Output Analysis

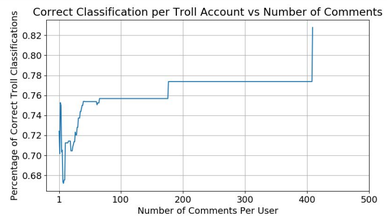


Figure 5: Troll Classification Acc. vs # Troll User Comments

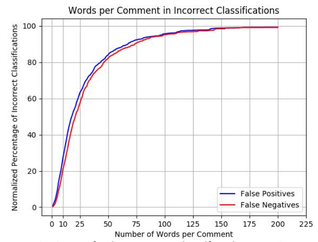


Figure 6: Cumulative Acc. Distribution vs Comment Size

Example	Test Sentence	Prediction	True
1	*Hillary won the popular vote*	1	0
	Sam won the popular vote	0	0
	Cats won the popular vote	1	0
	Hillary **played** the popular vote	0	0
2	*Wow I hope that's not me*	0	0
	Cats Wow I hope that's not me	1	0
3	*Hillary is 68, she just can't be all healthy*	1	1
	Hillary is 68, **he** just can't be all healthy	1	1
	Hillary is 68, she just can't be all **there**	1	1
	Hillary is 68, **he** just can't be all **there**	0	1

Figure 7: Classifications of Original and Modified Comment

- The model **performs better** at classifying troll accounts **with more comments** in the dataset
- The model also **performs marginally better** at classifying comments **with more words**, and is equally bad at misclassifying trolls and non-trolls on this metric alone
- **Trigger words caused classification to change** (e.g. Hillary). Some were dependent on certain context words being present

Conclusions

- **Summary:**
  - Both CNN and RCNN models classify Trolls with > 84% AUC score, significantly outperforming both baselines
  - Troll detection on Reddit seems viable!
- **Limitations:**
  - Our dataset is very limited
  - Dataset distribution is skewed
  - Model responds too aggressively to "triggers"
- **Future Work**
  - Model adjustments based on error analysis
  - Significant and rigorous data collection

References:

1. Devlin, J., Chang, M., Lee, K., \& Toutanova, K. (2018). A New Pre-Training Method for Training Deep Learning Models with Application to Spoken Language Understanding. CoRR.
2. Huffman, S. (2018, April 10). R/announcements - Reddit's 2017 transparency report and suspect account findings. Retrieved from https://www.reddit.com/r/announcements/comments/8bb85p/reddits_2017_transparency_report_and_suspect/
3. Punturo, B. (2019, January 08). Predicting Russian Trolls Using Reddit Comments – Towards Data Science. Retrieved from https://towardsdatascience.com/predicting-russian-trolls-using-reddit-comments-57a707653184
4. Lai, S., Xu, L., Liu, K., \& Zhao, J. (2015). Recurrent Convolutional Neural Networks for Text Classification. AAAI.