# Defending Against Adversarial Synonym Attacks

Enhancing the RobEn Framework to Protect NLP Models

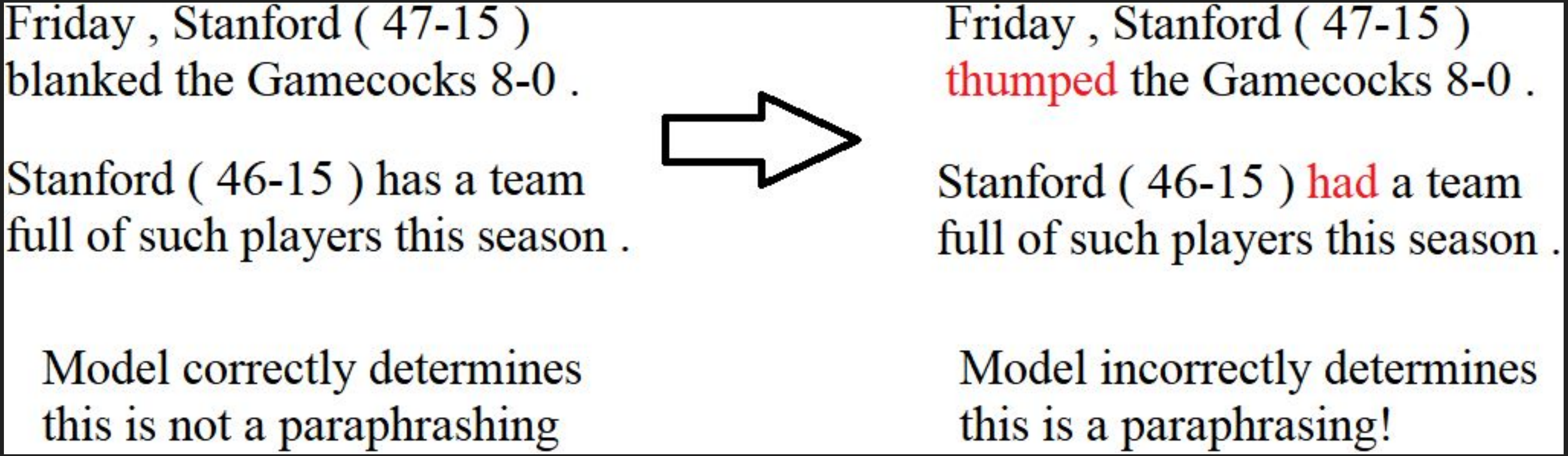
Sahil Farishta

EECS 598 Presentation
04/30/2021

# Motivation

- Adversarial attacks against NLP systems are getting increasingly strong
    - Recent attacks such as TextFooler [1] are able to cripple models
- NLP models are becoming increasingly prevalent in society
    - Conversational chatbots for shopping, banking, and even medical advice
    - Being able to fool or attack these models can have devastating consequences
- Defenses have not kept up with attacks
    - Most focus on preventing a single class of attack

# Motivation - Example




- Dataset consists of two pairs of sentences
  - Model needs to determine if they are paraphrases of each other
- Just by making the two substitutions highlighted in red, we can change the classification the model outputs!

# Robust Encodings (RobEn)

- The Robust Encodings (RobEn) [2] paper protected against typo attacks
  - An adversary might change prediction by creating a typo in a word
    - Substituting in ated instead of ate in the following sentence:
      - “The aunt ate the food” -> “The aunt ated the food”
    - May change classification from NLP model
  - Solution: Cluster words that are typos of each other
    - Words in the same cluster receive the same GloVe encoding
      - GloVe encoding [3] is the vectorized representation the NLP model sees
  - Performed well against state of the art typo attacks
  - Able to sit on top of any existing NLP model
- We use RobEn as a base and add synonyms to the clustering process
  - Desired goal: have a module to make any model robust against adversarial attacks

# Work Performed

- We extend the RobEn framework to generate clusters using synonyms
  - Determine synonyms for each word using WordNet [4]
    - Creates SynSets which consist of synonyms for any given word
- We demonstrate the accuracy of 4 different models trained in this manner
  - Each of the 4 models look at different ways of filtering what gets added to clusters
  - Run TextFooler attack on all models to show post attack accuracy

# Architecture Setup

- Our inputs are sets of sentences $X=\{X_1,X_2,\ldots,X_N\}$
  - Correspond with labels $Y=\{Y_1,Y_2,\ldots,Y_N\}$
- Our model is a function g: Z->Y where Z is the encoding domain
  - We represent the encoding function as α: X->Z
  - Usually use an embedding space like GloVe to represent Z
  - Classification becomes $\hat{y}=g(\alpha(X_i))$
    - Correct if $\hat{y} = y_i$
    - Shorthand for encodings each word of $X_i$ using α
  - Goal of the clustering algorithm: help alter α to provide robustness against synonym attacks

# Algorithm

- Start with an graph that contains nodes for each of the words in our vocabulary
  - No edges yet
- For each word in our vocabulary
  - Generate the synonyms for the word using WordNet SynSets
  - Add an undirected edge between the word and each of its synonyms
- If word $W_i$ and word $W_j$ share an edge in the clustering graph
  - Then $\alpha(W_i)=\alpha(W_j)$
  - Words that share an edge will be mapped to the same embedding
    - Can be suboptimal as large synonym chains can appear where two words will be clustered together even though they themselves are not synonyms
    - Agglomerative clustering technique presented in RobEn paper that helps alleviate this
      - Computationally very expensive however

# Example - Original Text

In midafternoon trading, the Nasdaq composite index was up 8.34, or 0.5 percent, to 1,790.47

# Example - Encoded RobEn + Synonyms

the the the, the nasdaq the index the the the, the 0.5 the, the the

# Example - Synonyms

a midafternoon a, the nasdaq complex a a a a, a 0.5 a , to a

# Example - Synonyms No StopWords

in midafternoon be, the nasdaq complex be was up be, or 0.5 be, to be.

# Experiments

- Evaluate using the MRPC dataset from GLUE [5]
- Ran experiment on 6 different models
  - Base BERT
  - RobEn model from original paper
  - Model that clustered all synonyms for each word from WordNet
  - Model that clustered top 3 synonyms for each word from WordNet
  - Model that clustered all synonyms for each word excluding stopwords from WordNet
  - Model that clustered top 3 synonyms for each word excluding stopwords from WordNet
- 2 tests performed for each model
  - Base accuracy in non-adversarial setting
  - Accuracy after running TextFooler attack

# Results

| Model | Normal Accuracy | Accuracy After TextFooler Attack |
|---|---|---|
| Base BERT | 0.877 | 0.152 |
| RobEn BERT | 0.809 | 0.189 |
| Synonym Encoded BERT | 0.755 | 0.6716 |
| 3 Synonym Encoded Bert | 0.745 | 0.6985 |
| Stopwords Filtered Synonym Encoded Bert | 0.7525 | 0.6446 |
| 3 Stopwords Filtered Synonym Encoded Bert | 0.7745 | 0.5980 |

- We see that the original two models perform very poorly against the attack
  - No synonym based defenses
- All new models achieve accuracy around 75% in a non-adversarial setting
  - Slightly lower than the RobEn model which is lower than the base BERT
- All new models are resistant to the TextFooler attack

# Results - Continued

- Test dataset is unbalanced (70% of examples have label 'True')
  - Can use confusion matrix to ensure that models are not exploiting dataset bias to perform well
- Look at confusion matrix for the top two models to evaluate
  - 3 Synonym Encoded BERT had bias
    - Very few false predictions
  - Stopword Filtered Synonym Encoded BERT had less bias with a better distribution
    - Lower accuracy but probably better ability to generalize to new input

| | Predicted True | Predicted False |
|---|---|---|
| Label True | 231 | 48 |
| Label False | 75 | 54 |

Confusion Matrix for 3 Synonym Encoded BERT

| | Predicted True | Predicted False |
|---|---|---|
| Label True | 157 | 122 |
| Label False | 23 | 106 |

Confusion Matrix for Stopword Filtered Synonym Encoded BERT

# Results - Discussion

- Models are able to resist the TextFooler attack
  - Previous work has shown it can cripple models (such as the base BERT we see here)
  - Limiting the generated clusters to only having a few synonyms per word and removing stopwords all result in similar accuracies
  - Removing stopwords from the clustering process results in a better distribution of false positives and true negatives across the unbalanced dataset
- We only looked at synonym based clustering here
  - Clusters generated with both synonym and typo defense grow too large
  - In the future, smarter clustering algorithms could lead to combining the two defenses
- Defense assumes that cluster information is available to adversary
  - No attack can take advantage of this right now
  - In the future, design an attack that takes into account the clustering information to fully test the robustness of this defense

# Conclusion

- RobEn clustering defense can be extended to work on synonyms
    - With a better clustering algorithm, we could integrate both synonym and typo based defense into one defense module
    - Defense can sit on top of any NLP model that considers encoded sentences to output labels
- Results show that various flavors of this defense can achieve around 75% accuracy in a non-adversarial setting
    - Maintain high performance in the regular setting
    - Accuracies from 60-70% when defending against TextFooler
        - Much higher than regular BERT model

# References


1. Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? A strong baseline for natural language attack on text classification and entailment. Association for the Advancement of Artificial Intelligence, 2020.
2. Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. Robust encodings: A framework for combating adversarial typos. Association for Computational Linguistics, 2020
3. Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. Empirical Methods in Natural Language Processing, 2014.
4. Princeton University. About wordnet, 2010.
5. A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding.In International Conference on Learning Representations, 2019