# Determining Political Issue Polarity with BERT

Neha Kumar and Sayan Das

W266 Section 5 Saturday 10a PT

Prof. Sid Reddy

# Research Goal and NLP

- 2020 Presidential Race saw many different candidates from diverse backgrounds
- Summarize candidates' stances on key issues for the 2020 election with NLP!

**Key Issues:**

- Pro / anti guns
- Pro / anti Medicare for all
- Pro / anti immigration
- Pro / anti abortion
- Pro / anti military spending
- Pro / anti tax on extreme wealth
- Pro / anti free higher education

Source

# Prior Research

**Studies classifying political stance**

- Classifying politician stance using tweets, by Johnson et al
  - Extracts temporal data
  - Supplemented with party affiliation
  - Bag of words type implementation

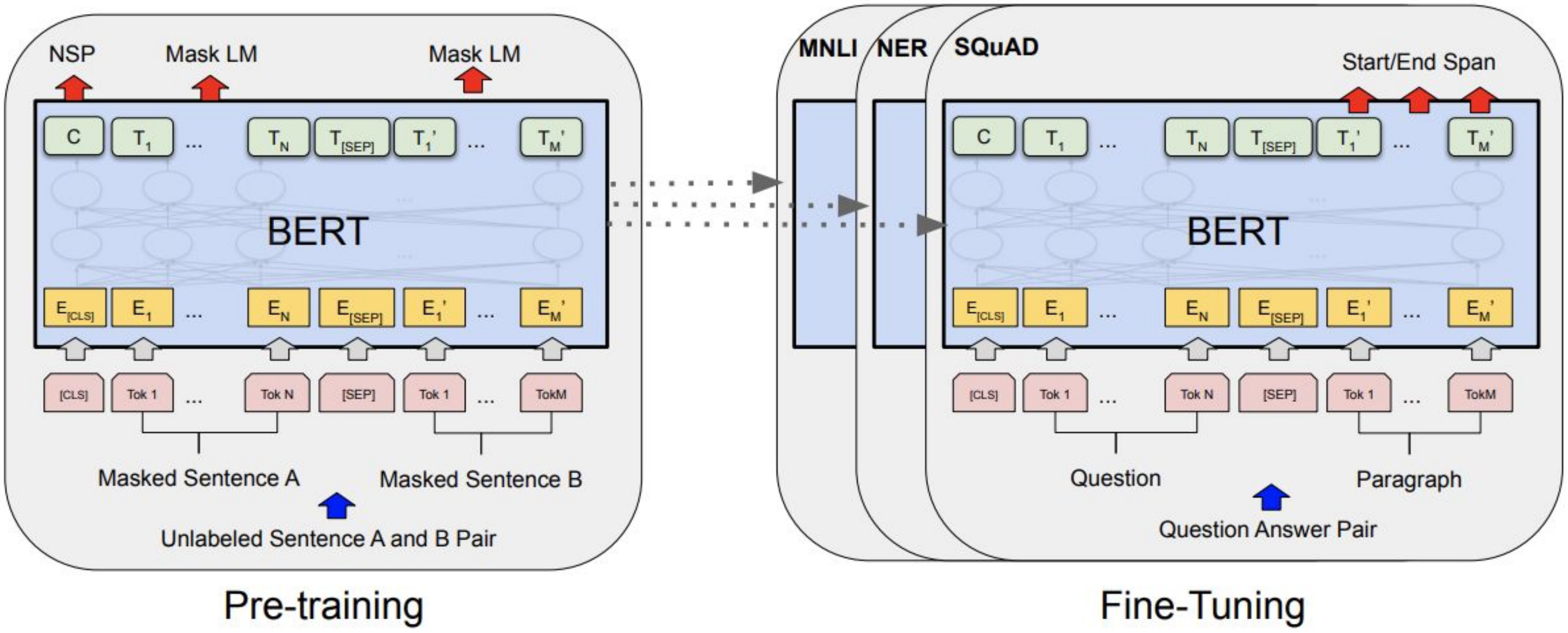
**Studies using BERT**

- Promising results shown in classifying toxic speech (by d'Sa et al)
- Improving performance with pretraining on unsupervised data (Han et al)
- Some studies found that the LSTM to still be preferable due to the efficiency to train that as opposed to BERT (110M parameters) - Adhikari et al

**To the best of our knowledge, there are no studies that apply BERT sequence classification on political text to classify candidate polarity on issues**

# BERT Architecture

**Used: bert-base-uncased**
12-layer, 768-hidden, 12-heads, 110M parameters.
Trained on lower-cased English text.




**Why BERT?**
Current SOTA, better than OpenAI GPT which is only unidirectional

Source

# Data Collection

**Round 1 Unsupervised Pretraining (D1 Data)**

Fake News Corpus

**Round 2 Unsupervised Pretraining (D2 Data)**

We wrote our own scraper to pull articles related to the 7 topics

**Labeled Training (D3 Data)**

We hand selected articles for and against each issue. We were able to label each sentence with the category since we knew the article it came from

**Test Data**

We wrote our own scraper to pull text from the Democratic candidates' NYT interviews and campaign websites

Trump tweets corpus

Icons from the noun project

# High-Level Workflow

## Pretraining

512 word segments

bert-base-uncased

Updated pretrained model

**Fine tune bert-base-uncased with 2 levels of unlabeled text data**

## Training

Sentence 1 | Label

Sentence 2 | Label

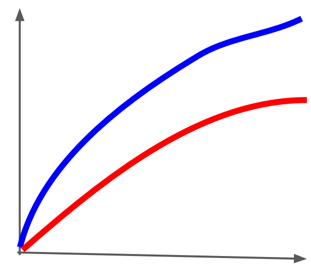
Sentence 3 | Label

Sentence 4 | Label

Sentence 5 | Label

**10% of sentences saved for validation**

## Validation



**Assess best hyperparameters**

## Testing

Sentence 1

Sentence 2

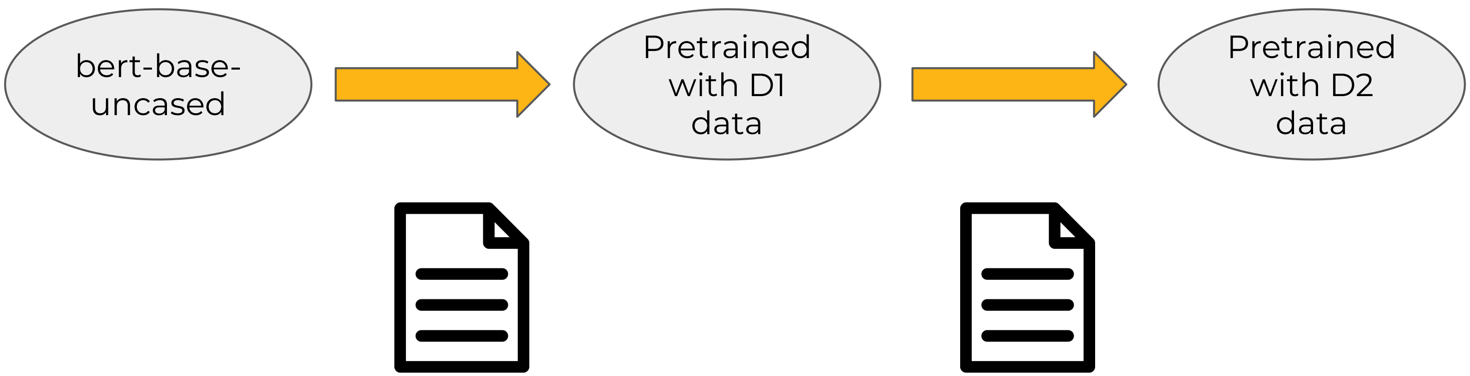
Sentence 3

Sentence 4

Sentence 5

**Predict label for candidate statements sentence by sentence and aggregate results per candidate**

# Closer look at pretraining




D1 data
(48M words)
~1 day of training

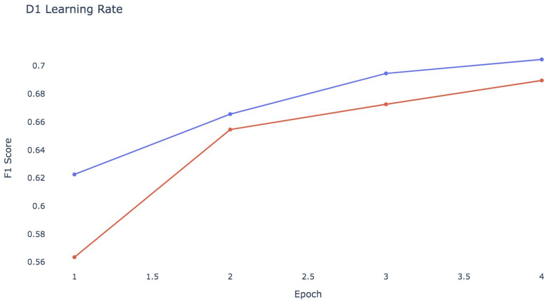
Segmented into
512 word chunks

D2 data
(1M words)
~20 min of training

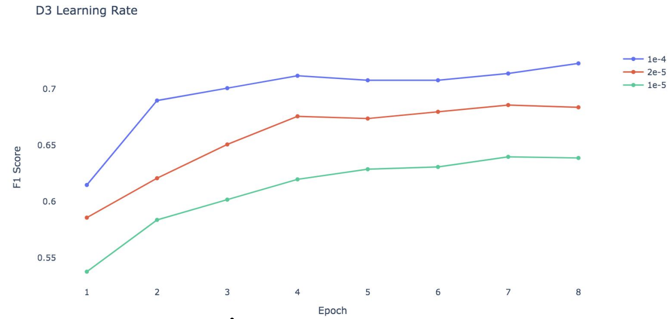
Segmented into
512 word chunks

# Hyperparameter tuning, showing F1 Score on Dev data

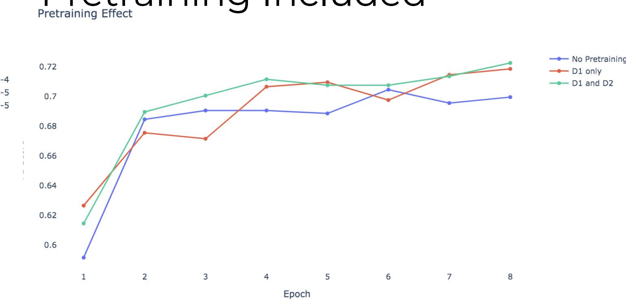
## D1 Learning Rate



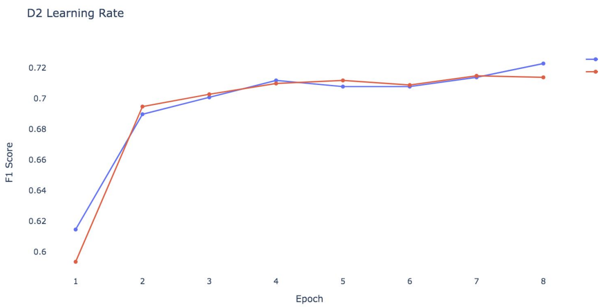

## D3 Adam Epsilon



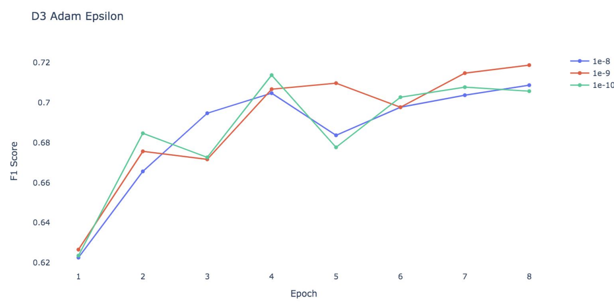

## Pretraining Included



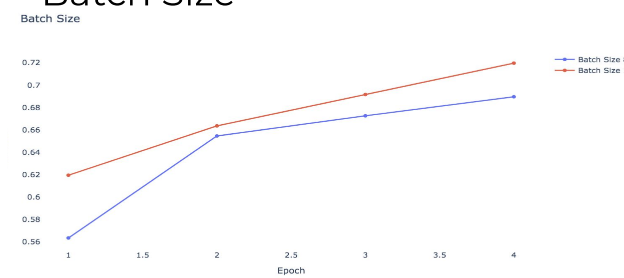

## D2 Learning Rate




## D3 Learning Rate

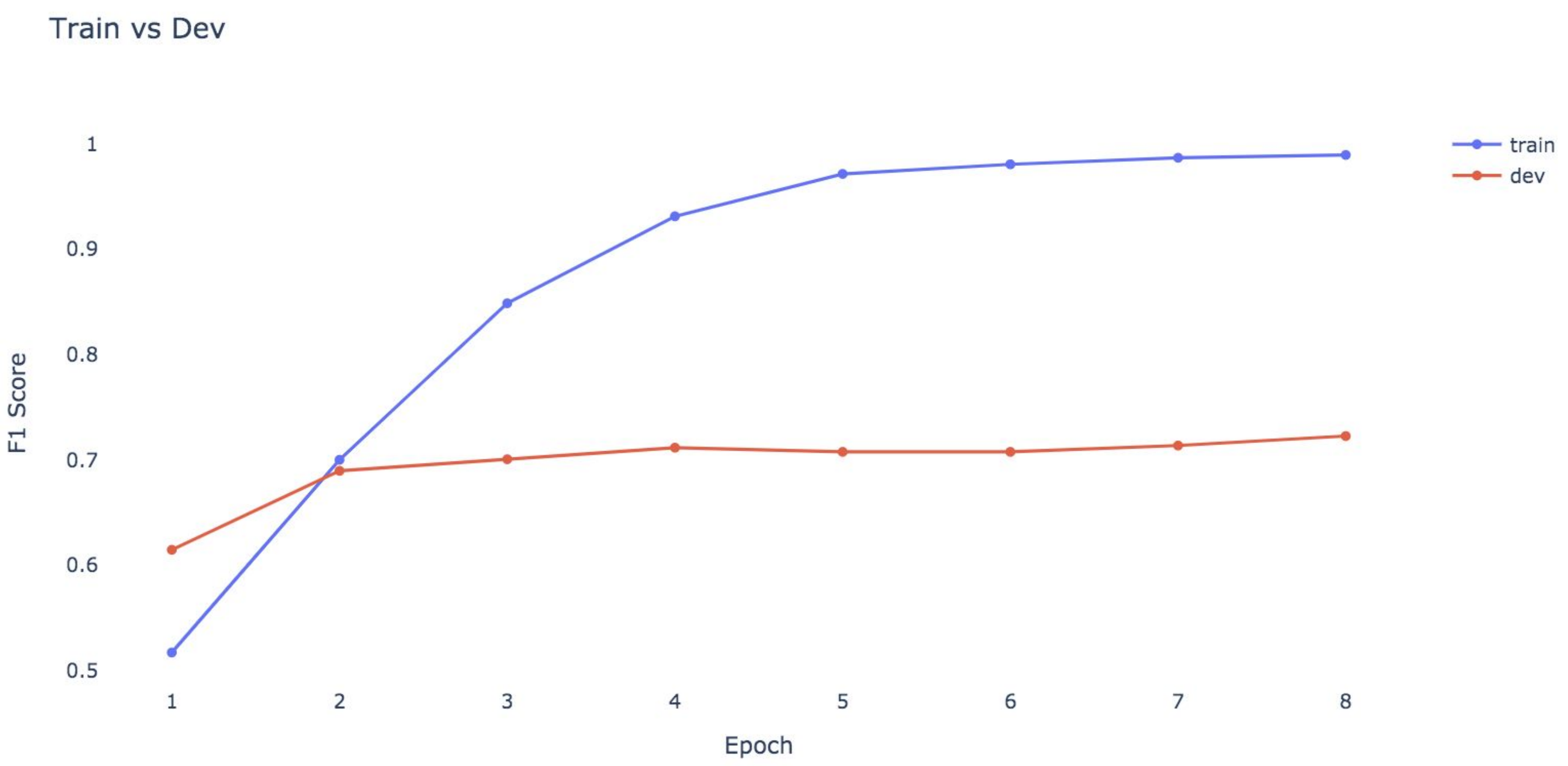



## Batch Size




**Best Performing Hyperparameters**
D1 and D2 Pretraining
Learning rate of 1e-4 for D1 D2 and D3
Adam Epsilon of 1e-9
Batch Size of 8

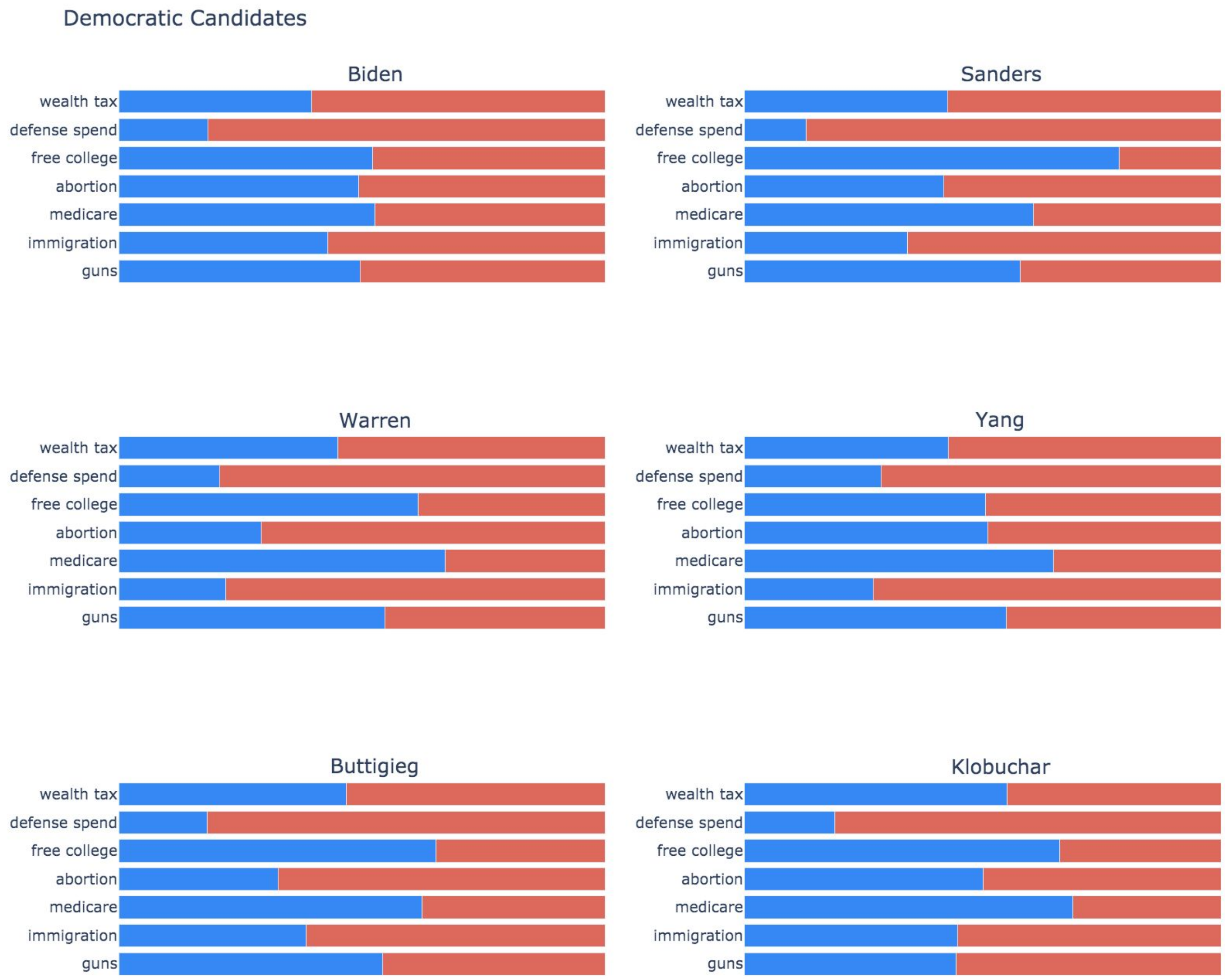
# Best Performing Model, Comparing Training and Dev F1



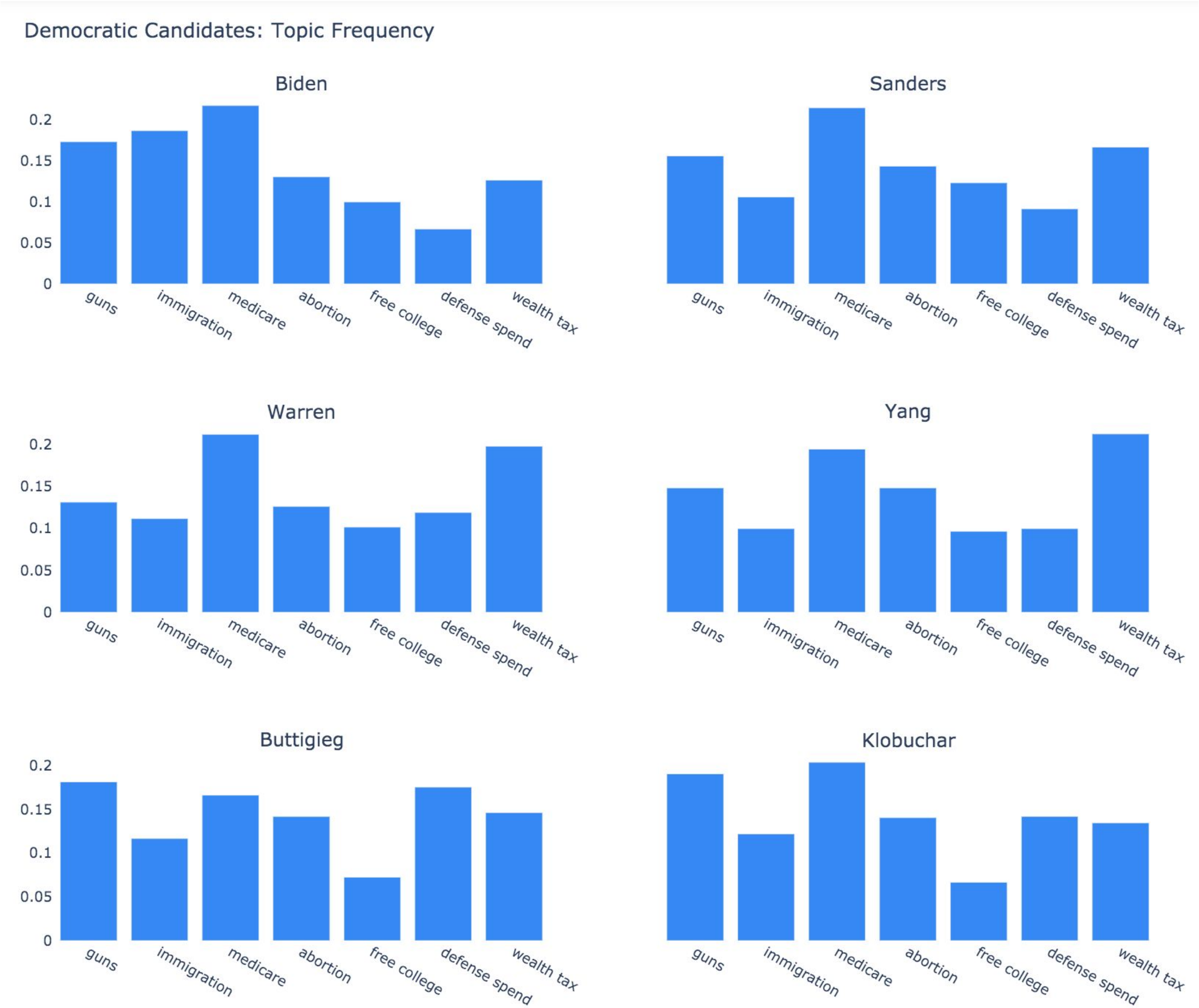
# Extremity Score Formula

$$\frac{1}{2} * \left( \frac{sum(pro) - sum(anti)}{sum(pro) + sum(anti)} + 1 \right)$$

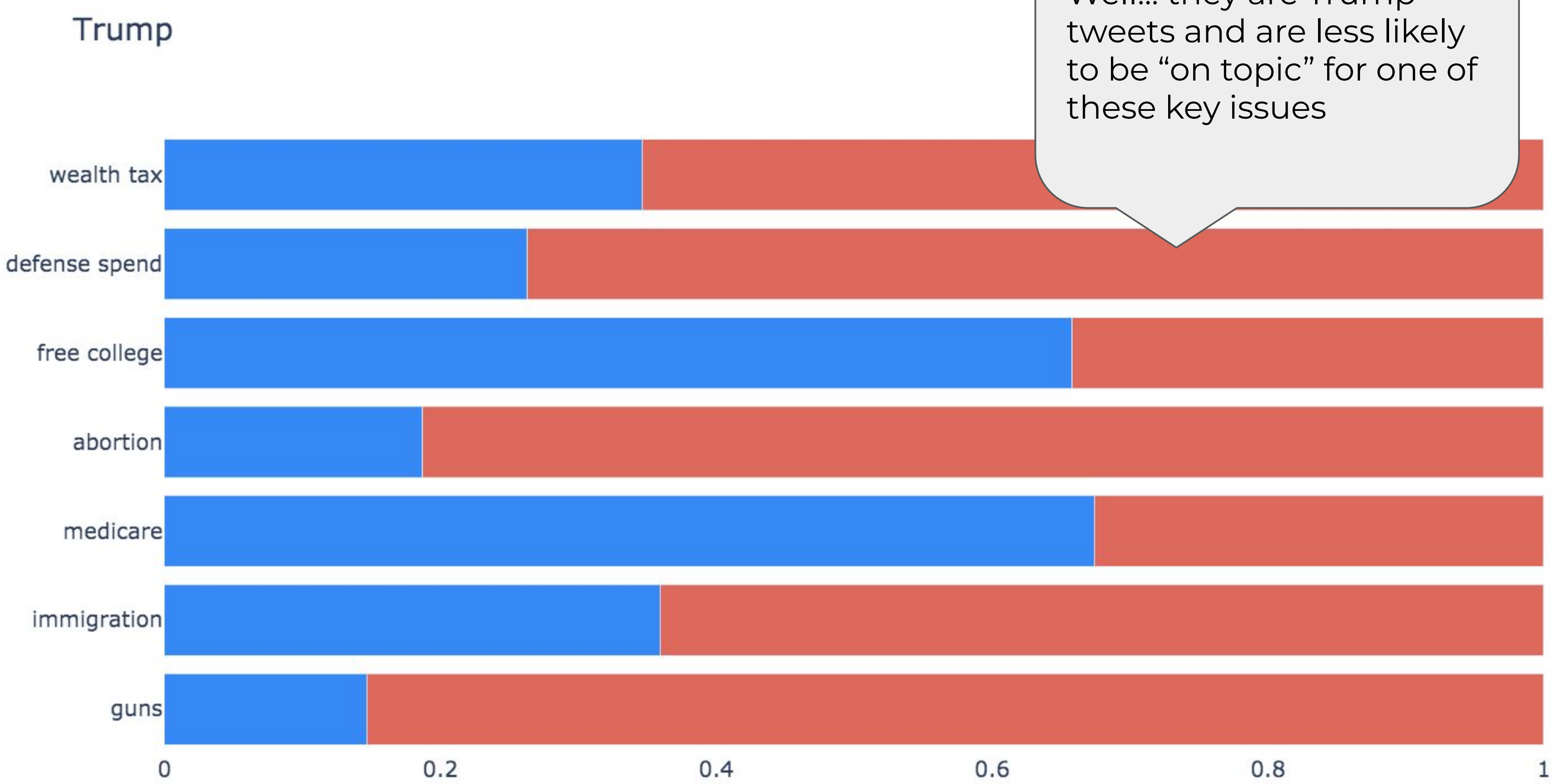
# Issue Polarity for Democratic Candidates



# Frequency of Statements For and Against each Issue



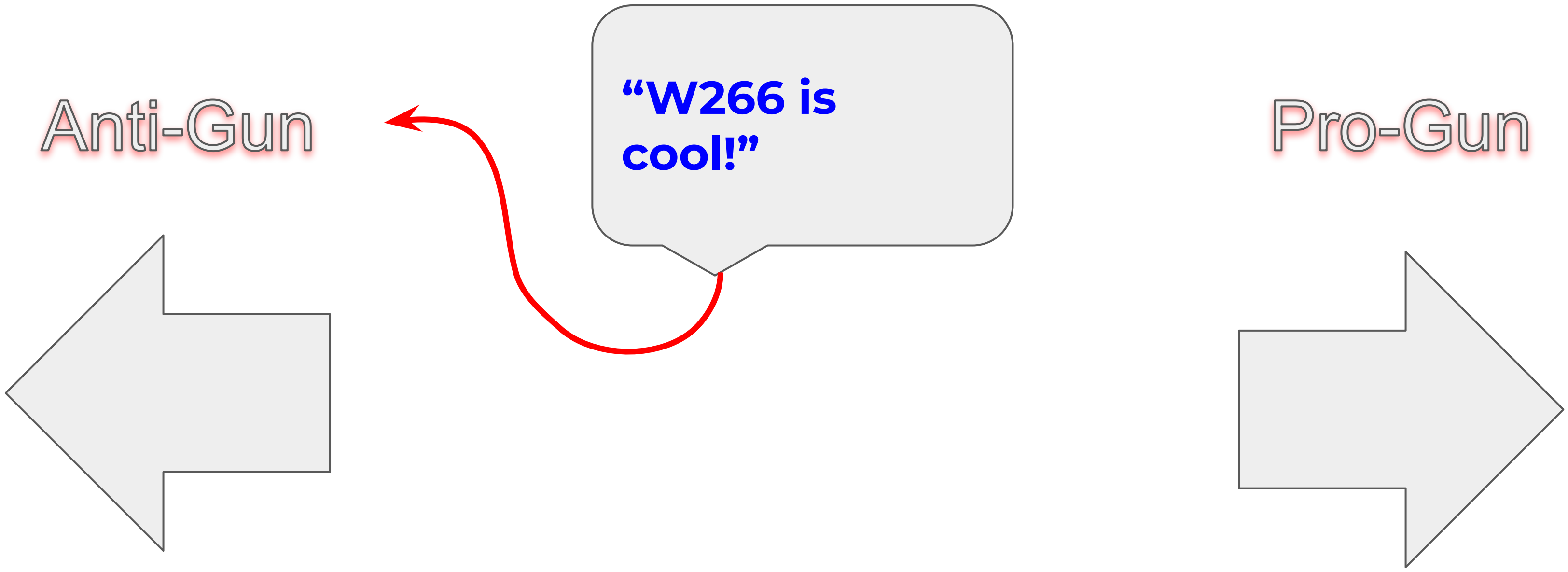
# Issue Polarity for Trump




Well… they are Trump tweets and are less likely to be “on topic” for one of these key issues

# Next Steps, Further Improvement

- More data! More pretraining!
- Address topic polarization
  - Multiple binary classification models



# Literature References

[1] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).https://www.aclweb.org/anthology/N19-1423.pdf
[2] Sun C., Qiu X., Xu Y., Huang X. (2019) How to Fine-Tune BERT for Text Classification?. In: Sun M., Huang X., Ji H., Liu Z., Liu Y. (eds) Chinese Computational Linguistics. CCL 2019. Lecture Notes in Computer Science, vol 11856. Springer, Cham https://arxiv.org/pdf/1905.05583.pdf
[3] Ashwin Geet d'Sa, Irina Illina, Dominique Fohr. BERT and fastText Embeddings for Automatic Detection of Toxic Speech. SIIE 2020 - Information Systems and Economic Intelligence, Feb 2020, Tunis, Tunisia. Ffhal-02448197 https://hal.inria.fr/hal-02448197/document
[4] Johnson, K., & Goldwasser, D. (2016). Identifying stance by analyzing political discourse on twitter. In Proceedings of the First Workshop on NLP and Computational Social Science. https://www.aclweb.org/anthology/W16-5609.pdf
[5] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Brew, J. (2019). Transformers: State-of-the-art Natural Language Processing. *arXiv preprint arXiv:1910.03771*. https://arxiv.org/pdf/1910.03771.pdf
[6] Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. (2019) DocBERT: BERT for Document Classification. David R. Cheriton School of Computer Science, University of Waterloo. https://arxiv.org/pdf/1904.08398.pdf
[7] Xiaochuang Han, Jacob Eisenstein. (2019). Unsupervised Domain Adaptation of Contextualized Embeddings for Sequence Labeling. Georgia Institute of Technology. https://arxiv.org/pdf/1904.02817.pdf

# Coding Resources

- https://github.com/huggingface/transformers
- For LSTM Baseline model: https://github.com/danwild/sagemaker-sentiment-analysis
- Tutorial for preprocessing: https://mccormickml.com/2019/07/22/BERT-fine-tuning/#3-tokenization--input-formatting
- For main training script:
  - https://github.com/huggingface/transformers/blob/master/examples/run_glue.py
  - https://aws.amazon.com/blogs/machine-learning/maximizing-nlp-model-performance-with-automatic-model-tuning-in-amazon-sagemaker/
  - https://github.com/danwild/sagemaker-sentiment-analysis/blob/163913a21837683e7605f6122ad2c10718347f65/train/train.py#L45
  - https://mccormickml.com/2019/07/22/BERT-fine-tuning/#3-tokenization--input-formatting