Twitter Bot Detection: Utilizing Transformers to Classify Tweets

Haley Farber, Michael Yazdani, Daniel Lampert
W266: Natural Language Processing with Deep Learning
UC Berkeley School of Information
{haleyfarber, michaelyazdani1997, danielclampert}@berkeley.edu

Abstract
Internet bot detection is an important challenge Twitter faces, and currently there is no policy banning Twitter bots. Using machine learning models and deep learning, Twitter should be able to flag bots on a tweet level. This project explores the use of BERT and DistilBERT models to classify if tweets were written by a bot or human. We also explore the use of LIME to identify which words infer a tweet came from an internet bot. This project's results show great promise in the use of deep learning models to detect internet bots as well as recommendations on ways to adjust these models to detect tweets written by bots with higher accuracy.

1.0 Introduction

In the era of social media and fake news, it has become increasingly important for social media sites to quickly identify fake accounts, in particular Twitter. A Twitter bot is an internet bot that operates from a Twitter account with automated tasks that could range from gaining social attention to spreading false political information. Twitter is one of the most popular social media sites and it is used for a variety of purposes, including accessing news. With the chaos from the 2020 general election in the US, Twitter amended their social media policy to regulate fake political information and false information in regards to COVID-19. Bot accounts play a major role in spreading false information on Twitter; however, Twitter does not have an explicit policy banning bots unless Terms of Service are violated by the account. The Twitter ecosystem and its easily accessible API makes Twitter vulnerable to bots.

With over 40 million bot accounts on Twitter (Chong, n.d.), regulating online bots is a must and necessity to build trust between real twitter users and the platform. Automatic bot detection with built in interpretability would allow Twitter employees or users to quickly identify bot accounts and see what portions of the tweet led to its identification as a bot. The built in interpretability would allow users to notice patterns in the language and syntax that bot accounts use.

To create a useful product we set a total of three goals: accuracy of predictions, speed of predictions, and interpretability of predictions. To achieve this we utilized both a pre-trained BERT Model and a DistilBERT model. We used 330,878 tweets to train and test our models. Using these models we were able to achieve an accuracy of 85.21% for BERT and an accuracy of 83.56% for DistilBERT. For BERT we had a precision of 94.6% and a recall of 78.6%. For DistilBERT we had a precision of 95.4% and a recall of 78.99%. Although DistilBERT suffered a slight accuracy decrease the training time was substantially sped up and the recall was actually higher. In production, the decrease in accuracy would be outweighed by the increased speed. Lastly, for BERT and DistilBERT local surrogates (LIME) was used to add interpretability to the predictions. LIME highlights the text that led to the prediction.

2.0 Related Work

One of the challenges of analyzing Twitter bots is discovering a large, labeled dataset of tweets where each tweet is marked as "human" or "bot". One of the more popular sites for Twitter bots is Botometer. Botometer is a project of the Observatory on Social Media, OSoME, at Indiana University. Botometer allows a user to input a Twitter username and returns a score determining if the account is from a bot or not. All of the datasets used to train the Botometer algorithm are publicly listed and linked on the site, creating a dataset repository for researchers studying Twitter bot detection. One of the datasets listed on the site is widely used by researchers: Cresci-2017.

Cresci-2017 is a compiled set of manually labeled tweets and Twitter account information. The tweets are categorized by the type of bot they are from, giving more insight to the intentions of the bots. Much work until 2019 on Twitter bot detection utilized this dataset.

One simpler approach utilizing this data that has yielded surprisingly accurate results is Support Vector Machines (Efthimion, et al., 2018). Using this method on bot identification has yielded accuracy as high as 95.77% accuracy on detection of a bot account (Efthimion, et al., 2018).

Kudugunta and Ferrara also utilized Cresci-2017 to determine if an account is from a bot or not as well as determining if a tweet is from a bot or not. For the task of determining if a Twitter account is human or not they achieved an accuracy of 99.81% . However, the task of determining if an individual tweet came from a bot or not proved to be much more difficult, with an accuracy of only 78% on the tweet data without data augmentation, and an accuracy between 88% and 90% using synthetic minority sampling in combination with ENN. However, by using an LSTM on the tweet data, the authors achieved an accuracy of 95% and creating their own contextual LSTM, an LSTM combining account metadata and tweet texta data, an accuracy of 96.33% (Kudugunta, et al., 2018).

Kudugunta and Ferrara's work led our team to believe that we'd need to run neural networks instead of simpler ML models to achieve optimal accuracy on tweet level bot detection. However, we noticed that most of the data from Cresci-2017 is outdated - the data ranges from 2009-2014. As a team, we decided to look into newer datasets with tweets labeled as written by bots or humans. We found the PAN19 dataset and papers running neural networks on tweet text data.

Färber, Qurdina, and Ahmedi were able to achieve an accuracy of 90.34% utilizing a CNN model on the PAN19 dataset and Przybyła obtained an accuracy of 91% using LASSO and/or BERT (Färber, et al., 2019).

Our group attempts to build upon the work of those utilizing the PAN19 dataset by not only also running state of the art models such as BERT, but by also creating mechanisms to improve interpretability.

3.0 Data

We used the PAN19 dataset, a corpus of tweets compiled and shared by the CLEF Initiative (Conference and Labs of the Evaluation Forum), a self-organized body whose main mission is to promote research, innovation, and development of information access systems (*PAN at CLEF 2019*, n.d.). The dataset contains over 26,000 accounts and 100 tweets per account. This dataset is maintained privately and is available by request for academic purposes. The data was formatted into individual XML files containing tweets for each account. The classification of each tweet as a bot or a human was provided in a separate XML file which had a file id. The data contains both English and Spanish tweets separated into different folders and we decided to just use English tweets. Though, these tweets may not be from this year, these accounts have been verified to be bots or humans.

We have organized the data into three columns: tweet_id, tweet, and classification. Tweet_ id indicates the unique Twitter user, tweet contains the text of a tweet, and classification labels a tweet as 0, written by a human, or 1,written by a bot. Our data consists of a training set corpus of 276,040 tweets, a development set corpus of 135,960 tweets, and a test set of 264,000 tweets. We used a one third split on the original training data to create the development set.

Overall, our tweet corpus consists of varied English language including Twitter specific syntax including URLs, emojis, @, # ,and RT. We created a bar chart of the most commonly used characters and words and found that @s are the most popular : mentioning another person in a tweet. Below is the distribution for the most popular tokens tweeted by humans and bot accounts.

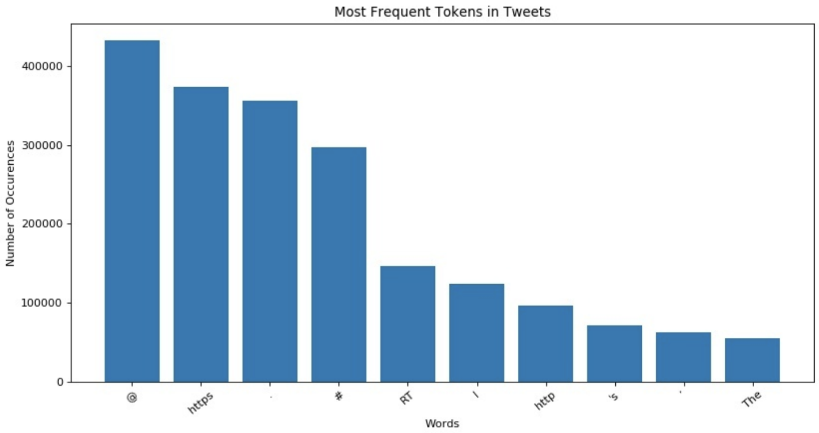


Fig. 1. Most Common Words in all Tweets (See appendix section 2 for more specific images)

4 Model

4.1 Baseline Models

In order to assess the relative gains of a more complex model, we decided to start with a variety of simpler models with several different encoding mechanisms.

4.1.1 Support Vector Machine

We applied multiple pre-processing techniques before encoding our text data. First we converted all text to lower-case, then we tokenized the text, and finally applied lemmatization. We used a Term Frequency-Inverse Document Frequency or TF-IDF encoding mechanism on the entire tweet corpus as the input for a Support Vector Machine Model (SVM). We decided to use TF-IDF because we believed that certain words would have a high frequency for bot accounts and others would have a high frequency for human accounts. For the SVM we used a linear kernel, a regularization strength of one, and a gamma of auto, meaning $\frac{1}{n}$ features were used. We decided to use a linear kernel rather than a polynomial kernel to both improve training speed and to see if the boundaries between bot and human tweets are linearly separable. With this model we achieved a surprisingly high accuracy of 75.6% on the test set. However, one major limitation of this model was the training time since the algorithm is not optimized to run on GPUs. In total it took 3 hours and 50 minutes to train.

4.1.2 Convolutional Neural Network

Before moving onto the state of the art natural language processing models, we decided to use a Convolutional Neural Network, CNN, with Word2Vec embedding. We zero padded the data to allow for the convolutional filters. To create the embeddings, we decided to use the Google News Word2Vec corpus since our corpus of tweets was not sufficiently large to build a representative vocabulary. We decided to use embeddings of length 300 which is sufficiently large to fit all tweets which have a character limit of 280.

Our CNN model has a total of three convolutional layers, three max pooling layers, and a sigmoid output layer. We trained the model over three epochs with a learning rate of 0.01. This model allowed us to reach an accuracy of approximately 78% on the test data. Even with GPU acceleration, this model still took nearly three hours to train.

4.2 Base BERT

For our transformer based model, we developed a BERT model, which stands for Bidirectional Encoder Representations from Transformers, a deep learning model where every output element is connected to every input element, and the weights between them are calculated based on connection. We trained on 3 epochs, with a dropout rate of 0.15, and a dropout rate of 15%. A diagram of the model is listed in figure below.

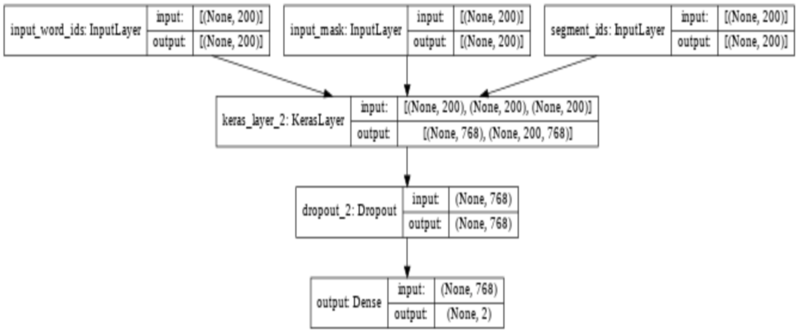


Fig. 2.   Base BERT Architecture

4.2.1 Base BERT Filtered Embeddings

We tried filtering out and replacing Twitter specific language and syntax, such as @ signs, hashtags, https links, and retweet signs, in tweets to analyze any difference in accuracy. These tokens are highly Twitter specific and thus we believed that removing and/or replacing them with customized strings would improve accuracy. Our decision to do this was also based on the fact that BERT was not explicitly trained on Twitter data.

4.2.2 Base BERT Parameter Updates

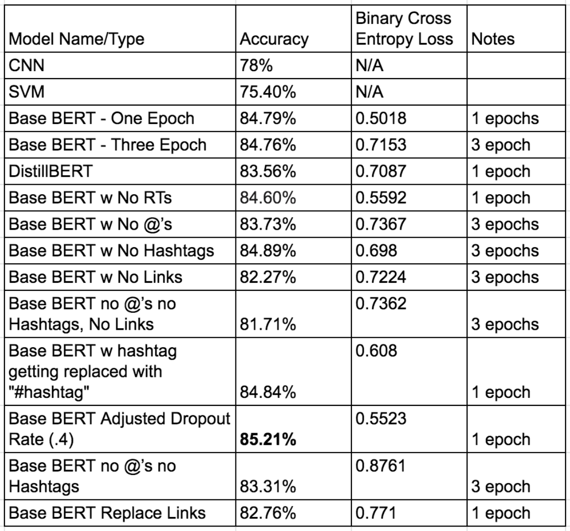
Another adjustment we made was updating the parameters of the BERT model throughout our iterations. Initially we trained the model using the standard practice of three epochs. However, in subsequent trials we noticed that training on one epoch does not reduce accuracy and sped up training time substantially. We also adjusted the dropout rate to 40% : a technique where randomly selected neurons are ignored during training. Using this high dropout rate improved overfitting on the training set.

4.3 DistilBERT

Due to the immense time (approximately five hours) it took to run our Base BERT model on three epochs, we decided to implement a DistilBert model that we believed to be more efficient and quicker. DistilBert is known for being a small, fast, cheap, and lighter transformer model- characteristics that make it more appropriate in a production environment.

5 Results

After developing multiple models, the best model we utilized was a BERT model - trained on one epoch with a dropout rate of .4. As shown in the table below, adjusting the BERT parameters improved our model slightly, but removing certain words from tweets proved unsuccessful in creating a large difference for our model's accuracy. We found that only removing hashtags improved our model, where as http link removal, @ sign removals, and 'RT' removals, only decreased the model accuracy overall.



Model Name/Type	Accuracy	Binary Cross Entropy Loss	Notes
CNN	78%	N/A	
SVM	75.40%	N/A	
Base BERT - One Epoch	84.79%	0.5018	1 epochs
Base BERT - Three Epoch	84.76%	0.7153	3 epoch
DistillBERT	83.56%	0.7087	1 epoch
Base BERT w No RTs	84.60%	0.5592	1 epoch
Base BERT w No @'s	83.73%	0.7367	3 epochs
Base BERT w No Hashtags	84.89%	0.698	3 epochs
Base BERT w No Links	82.27%	0.7224	3 epochs
Base BERT no @'s no Hashtags, No Links	81.71%	0.7362	3 epochs
Base BERT w hashtag getting replaced with "#hashtag"	84.84%	0.608	1 epoch
Base BERT Adjusted Dropout Rate (.4)	**85.21%**	0.5523	1 epoch
Base BERT no @'s no Hashtags	83.31%	0.8761	3 epoch
Base BERT Replace Links	82.76%	0.771	1 epoch

Fig. 3. Accuracy for all models

5.2 Discussion of Errors

5.21 BERT Errors

Error analysis permits us to see if there are any distinct patterns in the errors. For the BERT model, the majority of the errors were false negatives, meaning bot accounts were predicted to be human. There were a total of 33,903 instances of false negatives and a total of 7,176 false positives.

Confusion Matrix	Predicted : No	Predicted : Yes
Actual : No	124824	7176
Actual : Yes	33903	98097

Fig. 4. Confusion Matrix for BERT

Here are several examples of incorrectly classified tweets by the base BERT model trained on three epochs.

Error Type	**Tweet**
False Negative	That's the flag of Argentina: https://t.co/WvlSJ5HZy3 https://t.co/8Tv56CpOr7
False Negative	That's the flag of El Salvador: https://t.co/ep8CmNy7zj https://t.co/46lZ00DdYq
False Positive	Free from National Academies of Sciences: Mainstreaming Unmanned Undersea Vehicles into Future U.S. Naval Operations https://t.co/yAkCrGtDBs

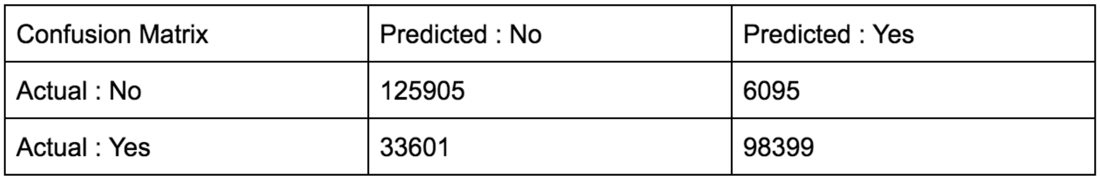
Fig. 5. Some incorrectly predicted tweets

One interesting thing we noticed about the false negatives is that many tweets had the form, “That's the flag of Argentina: https://t.co/WvlSJ5HZy3 https://t.co/8Tv56CpOr7”. These tweets were entirely mislabeled as human even though they came from bots. To specifically check the cause of this error, we checked how many times the string “That's the flag of …” occurred in the train data. We found that this string occurred zero times. However, tweets of this form occurred 1206 times in the test data signalling that at least 13 bots created this type of tweet. The fact that this tweet did not show up in the training data, but showed up numerous times in the test data contributed to this common error. Some potential causes of their mislabeling could be the fact that country names are strongly associated with human accounts. Another cause could be the fact that a portion of the URL for some reason is associated with human accounts.

Another commonality amongst misclassified tweets was the presence of URLs. Even though the conditional probability that a tweet is a bot given it has a URL or Pr(Bot|URL) is 65.7%, many tweets with URLs were incorrectly identified as human accounts. Likewise, many of the false positive results, or human accounts classified as bots, also contained URLs.

5.22 Distil-BERT Errors

The errors for DistilBERT were very similar to the errors in BERT. DistilBERT had a total of 33,601 false negative, or bot accounts identified as humans and a total of 6,095 false positives, or human accounts identified as bots. The results for DistilBERT were only slightly worse than BERT even though the training and prediction time was substantially sped up.



Confusion Matrix	Predicted : No	Predicted : Yes
Actual : No	125905	6095
Actual : Yes	33601	98399

Fig. 6. Confusion Matrix for DistilBERT

Similar to BERT, DistilBERT also misclassified all tweets of the form “That's the flag of Argentina: https://t.co/WvlSJ5HZy3 https://t.co/8Tv56CpOr7”. Due to the similarity in the errors between BERT and DistilBERT, we did not include incorrectly identified tweets.

6 Model Interpretability

6.1 Motivations

With many social decisions, reasoning is equally as important as the ultimate decision (Lipton, n.d.). Recent advances in machine learning have led to extremely accurate models, often at the expense of interpretability. As models become more and more prevalent in high stakes decision making, it has become increasingly important that model reasoning is considered (Lipton, n.d.).

In the area of natural language processing, model interpretability is of particular importance since language is more open to interpretation than many other machine learning applications (Lei, n.d.). However, the need for model interpretability has not been considered in the basic implementation of many of the most common and state of the art algorithms such as Convolutional Neural Networks and Transfer Learning. The lack of interpretability in these models has led to them being referred to as “black box models”.

6.2 Interpretability Methods

Most methods that attempt to add interpretability to deep neural networks have focused on the area of computer vision, however, the same concepts apply to text data. Most interpretability methods for deep learning attempt to find the neuron of a fully connected layer that is maximally activated (Molnar, 2021).

However, to add interpretability to our models, we decided to use model agnostic methods, in particular Local Surrogates (LIME) which work across all models.

Unlike other model interpretability methods, LIME focusses on local surrogates which means interpretability is given for individual predictions (Molnar, 2021). The process of local surrogates starts with choosing the instance of interest. In the case of Twitter bot prediction, the instance would be a tweet. Subsequently, the model is used to create a prediction for the instance of interest. Finally, an interpretable model such as Ordinary Least Squares is trained on the dataset and the prediction made by the blackbox model is explained using the simple model.

We implemented LIME for three of our algorithms, Convolutional Neural Networks (CNN), BERT, and DistilBERT to see if there are differences in the words or syntax that indicate a tweet came from a bot. For each model, LIME indicates the feature weight of each word in a tweet. In other words, LIME provides the probability that a given word comes from a bot or a human.

Although the models agreed on the vast majority of predictions, there were some interesting differences in the words that signaled if a tweet came from a bot or not. This difference was particularly evident in the incorrect predictions. Many words with a high probability of coming from a bot were relatively common words. One interesting example we noticed was that the word “UK” had a relatively high probability of coming from a bot. To see if this probability was logical, we calculated the probability of a tweet being a bot if it contains the word “UK”. The formula we used for this check was $P(Bot \mid "UK") \ = \ \frac{P("UK" \mid Bot)\, P(Bot)}{P("UK")}$. This turned out to be a probability of 30%. We tried this method on multiple words and found that the probability was correlated with the prediction, but it was not a direct relationship. This is likely the case because this Bayesian approach fails to account for context. For this reason, the word “UK” might be associated with a bot account given the context. To illustrate the output of LIME, we included images of LIME’s output for multiple tweets, including correct and incorrect predictions. The images are included in the appendix with the model type labeled. See Appendix section 1 to see images of LIME’s output for tweets.

6.3 Model Interpretability for Twitter Bot Identification: Use Case

Although Twitter’s policy does not explicitly ban bot accounts, understanding the specific characteristics that signal a tweet came from a bot would be useful in facilitating a more transparent social media environment (*Twitter’s Automation Development Rules | Twitter Help*, n.d.). Twitter potentially could incorporate interpretability by tagging tweets that are suspected to have come from a bot and by providing text highlighting to illustrate what information led to the tweet’s identification.

7 Conclusion and Future Research

This project demonstrates the ability of BERT and DistilBERT to correctly identify if tweets were generated by a bot or a human. We were able to experiment with different learning rates, number of epochs, dropout rate,and text preprocessing and its impacts on our model’s overall accuracy and loss. Finally, we developed a tool through LIME that Twitter could use in the future, to identify which words indicate whether a tweet was written by a bot or human. If we had more time, we would develop our project further in a few ways. First, we would want to train our model on a larger and more recent dataset.

Second, we would like to try integrated gradients optimized for transformer learning to see if the interpretability results vary from LIME.

As a product, we believe that our Twitter bot identification model coupled with interpretability mechanisms would allow Twitter employees to quickly identify bot accounts and to identify text and/or syntax that is associated with bot accounts. In the era of fake news, it is necessary for Twitter to address internet bots using deep learning and utilizing interpretability in order to gain insights on how bots are detected and to build trust between the platform and real users.

8. References

Chong, Z. (n.d.). *Up to 48 million Twitter accounts are bots, study says*. CNET. Retrieved April 10, 2021, from https://www.cnet.com/news/new-study-says-almost-15-percent-of-twitter-accounts-are-bots/

Cresci, Stefano & Pietro, Roberto & Petrocchi, Marinella & Spognardi, Angelo & Tesconi, Maurizio. (2017). The Paradigm-Shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race. 10.1145/3041021.3055135.

Färber, M., Qurdina, A., & Ahmedi, L. (2019). Identifying Twitter Bots Using a Convolutional Neural Network. *CLEF*.

Efthimion, P., Payne, S., & Proferes, N. (2018). Supervised Machine Learning Bot Detection Techniques to Identify Social Twitter Bots.

Kudugunta, S., & Ferrara, E. (2018). Deep Neural Networks for Bot Detection. *ArXiv, abs/1802.04289*.

Lei, T. (n.d.). *Interpretable Neural Models for Natural Language Processing*. 119.

Lipton, Z. C. (n.d.). In machine learning, the concept of interpretability is both important and slippery. *Machine Learning*, 28.

Molnar, C. (n.d.). *7.1 Learned Features | Interpretable Machine Learning*. Retrieved April 4, 2021, from https://christophm.github.io/interpretable-ml-book/cnn-features.html

Montavon, G., Samek, W., & Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, *73*, 1–15. https://doi.org/10.1016/j.dsp.2017.10.011

PAN at CLEF 2019. (n.d.). Retrieved April 10, 2021, from https://pan.webis.de/clef19/pan19-web/

Przybyła, Piotr. (2019). Detecting Bot Accounts on Twitter by Measuring Message Predictability.

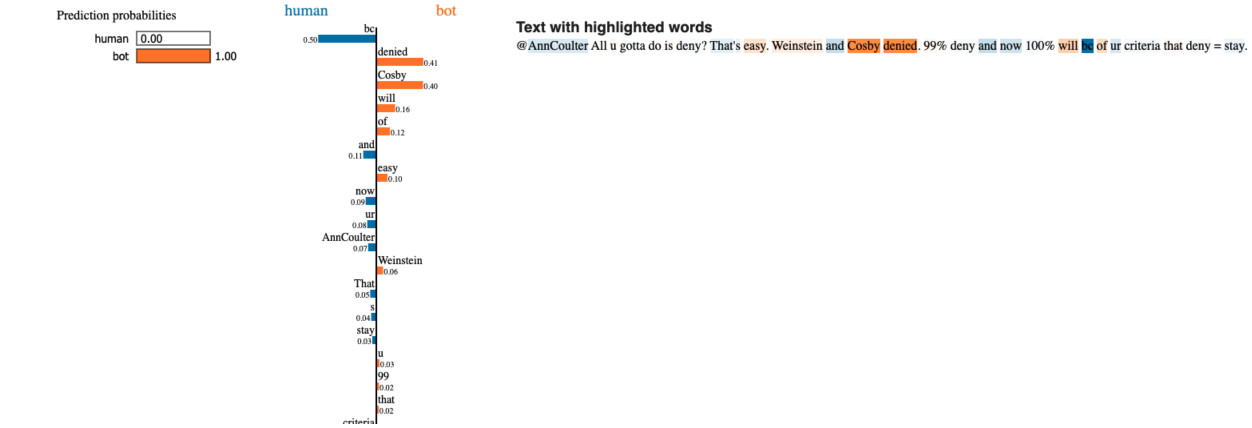
Twitter's automation development rules | Twitter Help. (n.d.). Retrieved April 3, 2021, from https://help.twitter.com/en/rules-and-policie/twitter-automation

9. Appendix

Section 1: LIME Images

Tweet 1

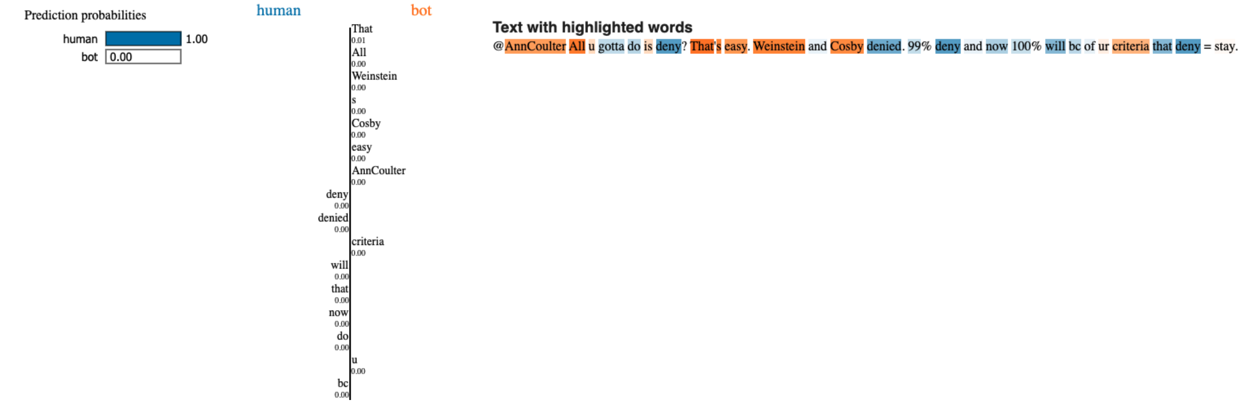
Model Type	Correct Prediction	Type of Account
CNN	No	Human



Text with highlighted words

@AnnCoulter All u gotta do is deny? That's easy. Weinstein and Cosby denied. 99% deny and now 100% will bc of ur criteria that deny = stay.

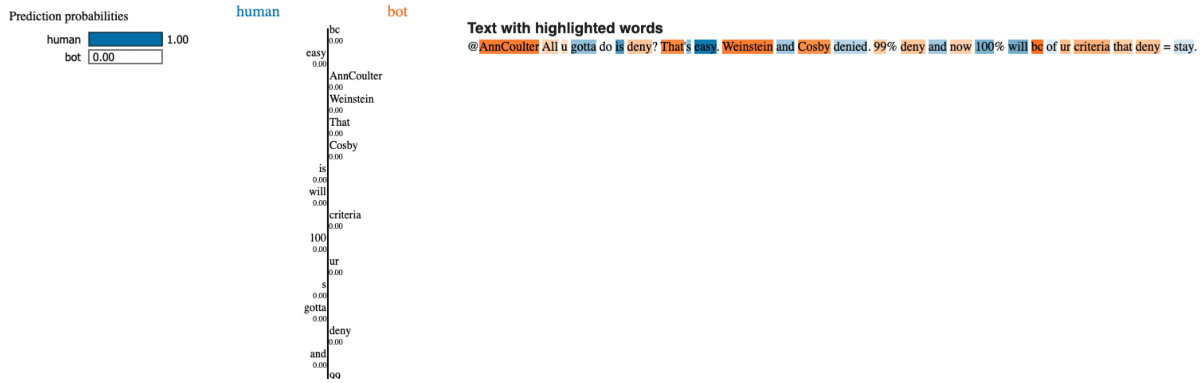
BERT	Yes	Human



Text with highlighted words

@AnnCoulter All u gotta do is deny? That's easy. Weinstein and Cosby denied. 99% deny and now 100% will bc of ur criteria that deny = stay.

Distil-BERT	Yes	Human

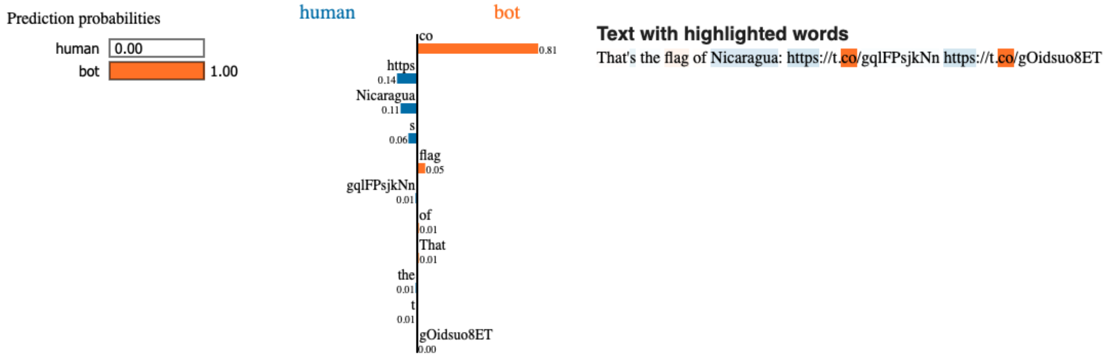


Text with highlighted words

@AnnCoulter All u gotta do is deny? That's easy. Weinstein and Cosby denied. 99% deny and now 100% will bc of ur criteria that deny = stay.

Tweet 2

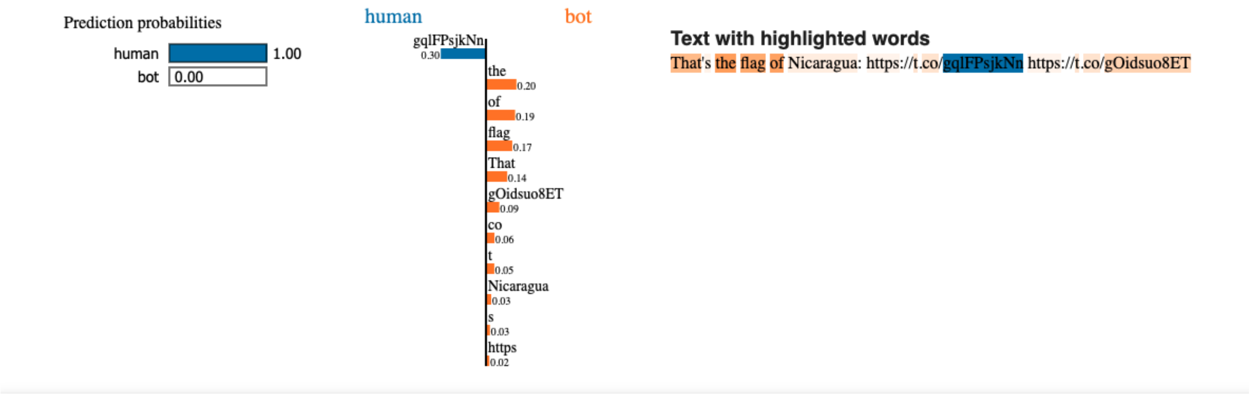
CNN	Yes	Bot



Text with highlighted words

That's the flag of Nicaragua: https://t.co/gqlFPsjkNn https://t.co/gOidsuo8ET

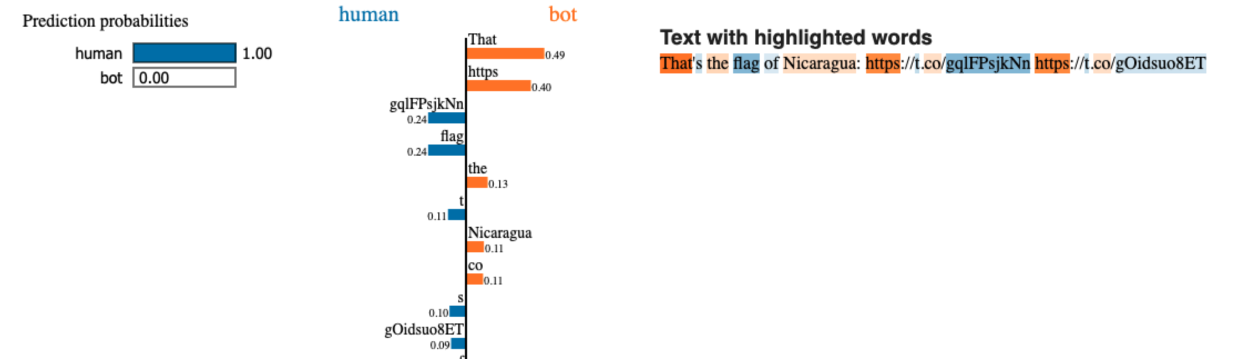
BERT	No	Bot



Text with highlighted words

That's the flag of Nicaragua: https://t.co/gqlFPsjkNn https://t.co/gOidsuo8ET

Distil-BERT	No	Bot

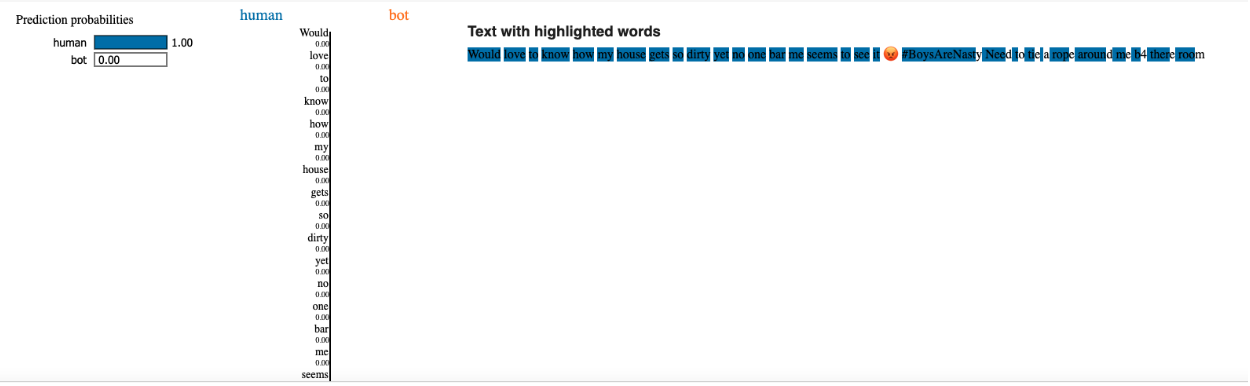


Text with highlighted words

That's the flag of Nicaragua: https://t.co/gqlFPsjkNn https://t.co/gOidsuo8ET

Tweet 3

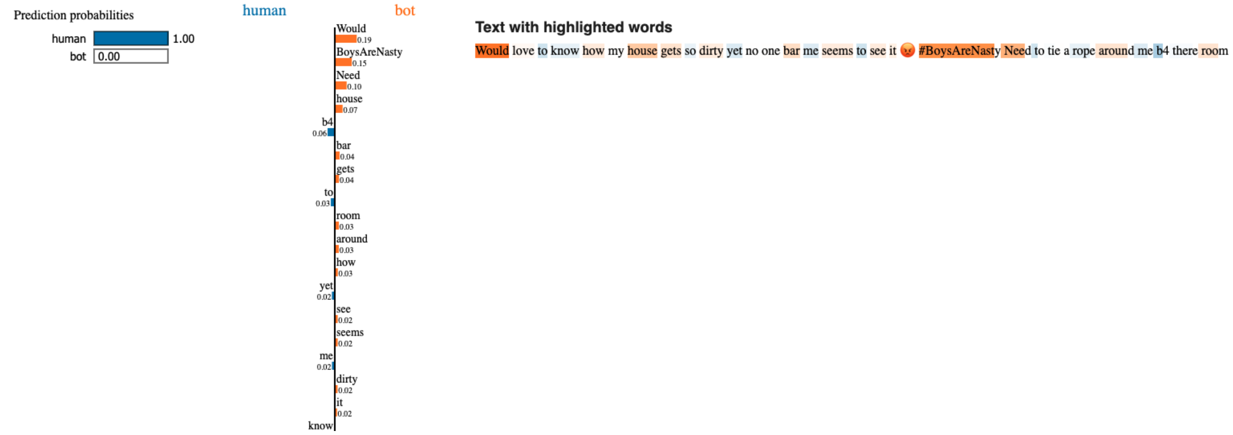
CNN	Yes	Human



Text with highlighted words

Would love to know how my house gets so dirty yet no one bar me seems to see it 😡 #BoysAreNasty Need to tie a rope around me b4 there room

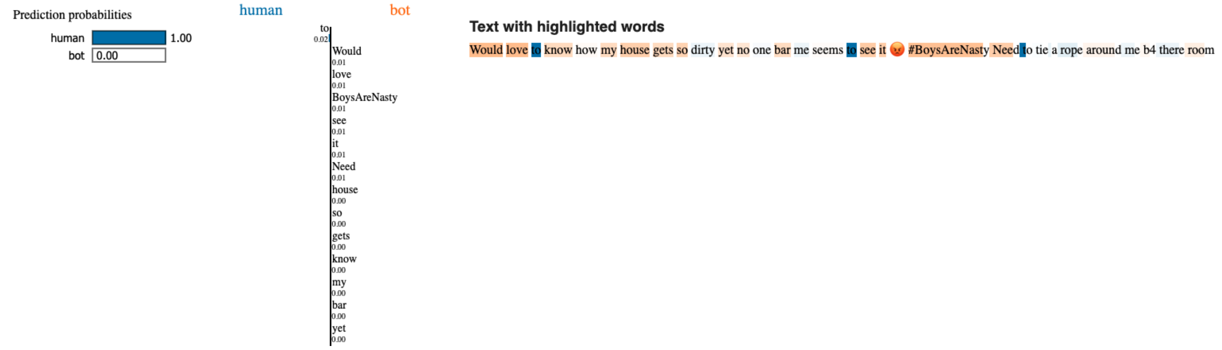
BERT	Yes	Bot



Text with highlighted words

Would love to know how my house gets so dirty yet no one bar me seems to see it 😡 #BoysAreNasty Need to tie a rope around me b4 there room

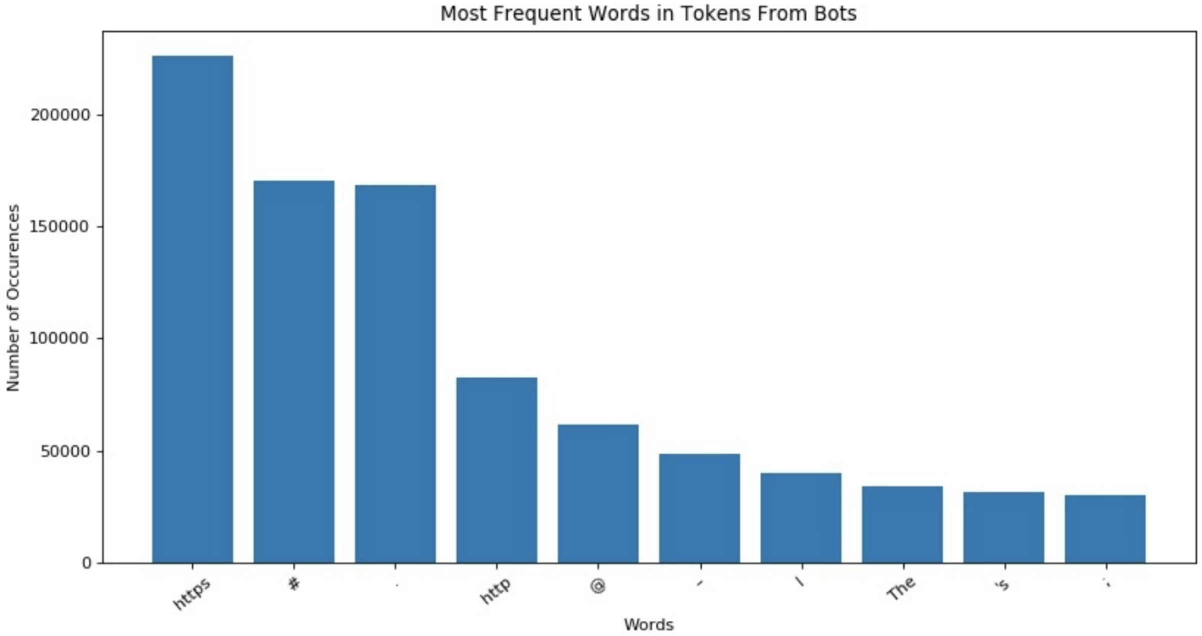
Distill-BERT	Yes	Bot



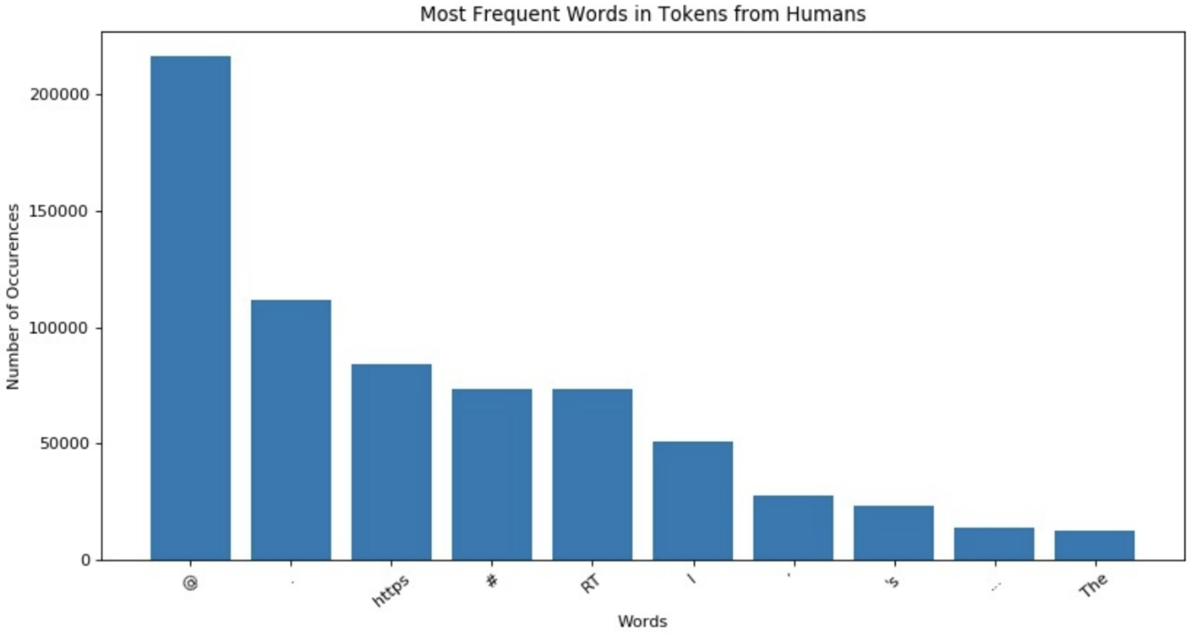
Text with highlighted words

Would love to know how my house gets so dirty yet no one bar me seems to see it 😡 #BoysAreNasty Need to tie a rope around me b4 there room

Section 2: Most Common Words and Tokens in specific portions of data



Section 2: Fig. 1. Most common tokens in tweets from bot accounts



Section 2: Fig. 2. Most common tokens in tweets from human accounts