Classifying and Explaining Fake News Using Graph Attention Networks and LIME Analysis of BERT Embeddings

Marcus A. Streips
UC Berkeley / MIDS
mastreips@berkeley.edu

Theodore Fong
UC Berkeley / MIDS
theodorefong@berkeley.edu

Abstract

A method combining graph attention networks (GATs) with local surrogate models of BERT embeddings is proposed. Graph relationships between news articles (nodes) are built using an adjacency matrix of user behavior and are weighted using GAT model edge attentions. BERT Embeddings are generated from a corpus of Fake and Real News Articles (Buzzfeed) and the predictions of a BERT Classifier model on interesting nodes are explained using Local Interpretable Model-Agnostic Explanations (LIME).

1 Introduction

Despite the headlines of Fake News from the 2016 Presidential election, building an acceptable fake-news detector for the pending 2020 Presidential election is still a challenge for the data science community. Approaches to identifying fake news include: sentiment analysis, text semantic analysis, network path propagation, image classification, social network analysis, and 'hybrid' models that combine some variation of the listed approaches. Although it is critical for models to be able to classify a fake news article, it is equally as critical for a model to bring transparency on explaining 'why' an article is labeled as 'fake news'.

Text classification alone is not sufficient to detect fake news and should be supplemented with other modeling techniques to be effective. We propose a combination of text and graph analysis with explainability to better understand how to detect the spread of fake news via its content and its relationship to other news articles.

2 Background

The use of social network data to detect fake news is an ongoing field of research. Yang et al. used users' opinions towards the authenticity of a news tweet on social media to build an unsupervised fake news detection framework (UDF) using a generative Bayseian network model and collapsed Gibbs sampling. They reported that this technique yielded an accuracy of 0.759 and F1-score of 0.741 using the Buzzfeed data as a baseline (Yang et al., 2019).

Shu et al., explore leveraging the correlations between publisher bias, news stance and user engagements in tri-relationship fake news detection framework (TriFN). This framework is a semi-supervised linear classification technique using user-user social relationships, user credibility and new user engagement scores, with a reported accuracy of 0.864 and F1-score 0.870 using the Buzzfeed data as a baseline (Shu et al., 2017).

Shu et al. also propose a method of explainable fake news detection using a news content encoder, a user comment encoder, a sentence-comment co-attention component and a fake news prediction component. (Shu et al., 2019). They use a bi-directional gated recurrent unit (GRU) as the encoder and the attention weights from the encoder attention vectors to help predict whether and article is fake or real. Shu et al. report that this technique yielded an accuracy of 0.808 and F1-score of 0.755 using the GossipCop data as a baseline. (Shu et al., 2019)

None of these methods explore using bi-directional encoder representations from transformers (BERT) embeddings (Devlin et al., 2019) as a content encoder or the use of graph attention networks (GATs) (Velickovic et al., 2018) for classifying and explaining fake news data as we present in this paper.

3 Methods

We present a semi-supervised, classification model that derives news-news relationships from user-news relationships on social media and feeds the news content as extracted BERT feature embeddings into a Graph Attention Network (GATs) to classify the news articles as real or fake. Using the graph information (nodes and attention-based edge weights) we identify interesting articles for further examination through the LIME framework to determine what features fake news articles have in common.

3.1 Data

Definition (Fake News). *Fake news is a social media posting, purporting to be news, that is verifiably false.*

The data used for this analysis is the Buzzfeed data from the Arizona State University Data Mining and Learning Lab (DMML).[1] Fake news ground truth are collected from the BuzzFeed platform. This dataset comprises of a complete sample of news published on Facebook from 9 news agencies over a week close to the 2016 U.S. election from September 19 to 23 and September 26 and 27. Every post and the linked article were fact-checked claim-by-claim by 5 BuzzFeed journalists. It contains 1,627 articles 826 mainstream, 356 left-wing, and 545 right-wing articles. From that dataset, 91 real and 91 fake news articles were randomly selected for this analysis. The data also identifies which articles was posted/shared by which user. The relationship between news articles can be derived through a common user, thus a news adjacency matrix was generated using the user/news information and fed into the GAT model as an edge list tuple.

3.2 Bi-directional Encoder Representations from Transformers (BERT) Embeddings

BERT is a multi-layer, bi-directionally trained transformer encoder stack. A transformer is an encoder-decoder architecture model which uses a multi-head attention mechanism to feed forward a tokenized word sequence to a decoder. The BERT encoders are trained using a masked language model, whereby 15% of words in a corpus are hidden to train the model in both left and right contexts, allowing the model to infer the positional information of a word in a sequence. The BERT encoder stack generates an output vector that can be used as the embedding input for any classifier, including GATs as discussed below.

[1]https://github.com/KaiDMML/FakeNewsNet/tree/old-version/Data

Using the "bert_uncased_L-12_H-768_A-12/1" pretrained tensorflow model, BERT word embedding extraction was performed on each Buzzfeed article in the data set with the resulting features (output vector) fed into a GAT classifier.[2]

3.3 Graph Attention Network (GAT)

The news adjacency matrix, BERT feature embeddings and input labels were split into training, validation and testing sets of 109, 36, 36 samples respectively and fed into a single-layer GAT model with 129 input features, 2 outputs (binary classifier), 4 attention heads, 8 hidden units and two drop-out layers (features/attention) while using negative-slope Leaky ReLU activation functions. The resulting GAT model had a total of 9752 edges, 182 nodes and 2 Classes. Hyperparameters were tuned using a randomized grid search. The resulting graph has the embedded features of the news articles as nodes or vertices and the edges as the relationship between each article. Attention weights are represented as edge weights in the graph. The GAT model architecture[3] is as follows:

```
  GAT(
(gat_layers): ModuleList(
  (0): GATConv(
    (fc): Linear(in_features=129, out_features=32)
    (feat_drop): Dropout(p=0.6)
    (attn_drop): Dropout(p=0.6)
    (leaky_relu): LeakyReLU(negative_slope=0.2)
  )
  (1): GATConv(
    (fc): Linear(in_features=32, out_features=2)
    (feat_drop): Dropout(p=0.6)
    (attn_drop): Dropout(p=0.6)
    (leaky_relu): LeakyReLU(negative_slope=0.2)
  )
 )
)
```

For a graph convolutional network (GCN), a graph convolution operation produces the normalized sum of the node features of neighbors.

$$h_i^{(1+1)} = \sigma \left(\sum_{j \in N(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)} \right) \tag{1}$$

where $N(i)$ is the set of its one-hope neighbors, $c_{ij} = \sqrt{|N(i)|}\sqrt{|N(i)|}$ is a normalization constant based on graph structure, σ is the activation function (ReLU), and $W^{(l)}$ is a shared weight matrix for node-wise feature transformation.

GAT introduces the attention mechanism as a substitute for the statistically normalized convolution operation. The equations for computing the

[2]https://github.com/google-research/bert (adapted from Chayapathy, et al. 2019)

[3]https://github.com/PetarV-/GAT

node embedding $h_i^{(l+1)}$ of layer $l+1$ from the embeddings of layer l:

$$z_i^{(l)} = W^{(l)} h_i^{(l)} \quad (2)$$

$$e_{ij}^{(l)} = LeakyReLU(\vec{a}^{(l)^T}(z_i^{(l)} \,||\, z_j^{(l)})) \quad (3)$$

$$\alpha_{(ij)}^{(l)} = \frac{\exp(e_{ij}^{(l)})}{\sum_k \in N(i) \exp(e_{ik}^{(l)})} \quad (4)$$

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} \alpha_{ij}^{(l)} z_j^{(l)}\right) \quad (5)$$

Equation (2) is a linear transformation of the lower layer embedding $h_i^{(l)}$ and $W^{(l)}$ is its learnable weight matrix. Equation (3) computes a pair-wise, un-normalized attention score between two neighbors, concatenating the z embedings of the two nodes, where $||$ denotes concatenation, then taking a dot product of it and a learnable weight vector $\vec{a}^{(l)}$, applying a LeakyReLU as the activation function as an additive attention model. Equation (4) applies a softmax to normalize the attention scores on each node's incoming edges. Equation (5) aggregates the neighborhood embeddings, scaled by the attention scores (Zhang, et al., 2019)(see Figure 1).

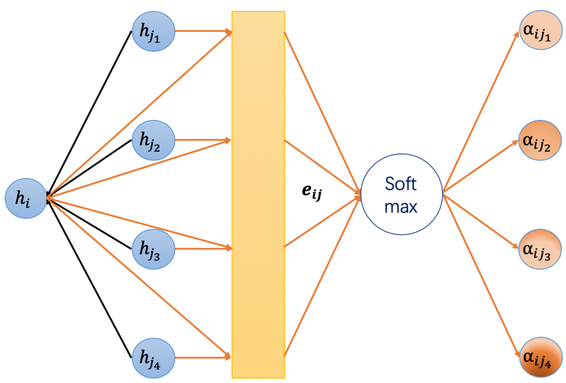


Figure 1: GAT Network (Zhang, et al., 2019)

Once the GAT model was trained, the edges with the heaviest attention weights were identified and their associated nodes were selected for further explanatory analysis using the LIME framework.

3.4 Local Interpretable Model-Agnostic Explanations (LIME)

LIME,[4] an algorithm that can explain the predictions of any classifier or regressor by approximating it locally with an interpretable model (Ribeiro, et al., 2016). It does so by "perturbing" random samples of data and observing the output of a model trained to approximate the local behavior of the model under scrutiny. This process includes removing words from a text classification model. LIME generates a new dataset consisting of permuted samples and the corresponding predictions of the black box model. On this new dataset LIME then trains an interpretable model, which is weighted by the proximity of the sampled instances to the instance of interest. The LIME model is expressed formally as:

$$\text{explanation}(x) = \underset{g \in G}{\text{argmin}}\, L(f, g, \pi_x) + \Omega(g) \quad (6)$$

The locally interpretable model (x) minimizes the loss (L) of how close the explanation is to the prediction of the original model (g) in the neighborhood defined by π. The model complexity $\Omega(g)$ is defined by the user by selecting the number of features the local model may use.

The interpretable model has local fidelity, meaning it is a good approximation of the black box model model locally. By selecting the number of features used in the interpretable model, one can balance interpretability versus model fidelity (the lower being more interpretable). The output of LIME is a list of explanations reflecting the contribution of each feature to the prediction of a sample. This enables us to determine which feature changes will have the most impact on prediction. The biggest disadvantage of LIME is the instability of its explanations which can vary widely in with repeated sampling (Alvarez-Melis, et al.)

The BuzzFeed news content from the nodes of interest was converted into tsv format and fed into a Bert Classifer model using a scikit-learn wrapper that allowed us to generate the binary class of prediction probabilities needed to feed into the LIME framework. The results and conclusions of the LIME analysis are discussed below.

4 Results and Discussion

4.1 Baseline BERT Classifer Model Performance

A binary BERT linear classifier model using the pretrained "bert_uncased_L-12_H-768_A-12/1" model was fit to the BuzzFeed training data, with 10% of the data held-out for validation and, 10% used as the warmup-proportion. The result-

[4] https://github.com/marcotcr/lime

ing baseline performance of the model after 10 epochs is reported in Table 1.

label	precision	recall	f1-score
negative	0.63	0.79	0.70
positive	0.77	0.61	0.68
accuracy			**0.6885**

Table 1: Bert Classifier Baseline Model Performance

This baseline BERT classifier was used in all subsequent LIME analysis.

4.2 GAT Model Performance

BERT embeddings were fed into a GAT model as features as described above. The model was trained for 10 epochs and the performance of the GAT model is reported in Table 2. We observe an average of 10% improvement of both accuracy and F1-scores over the baseline. The scores are consistent with what has been reported in the literature (Yang et al. 2019; Shu et al., 2019, Chayapathy, et al., 2019). The results support this paper's hypothesis that augmenting text classification models with graph relationships significantly improves the performance of models attempting to classify fake news articles.

label	precision	recall	f1-score
negative	0.79	0.80	0.78
positive	0.79	0.77	0.78
accuracy			**0.7889**

Table 2: GAT Model Performance after Hyperparameter Tuning

4.3 GAT Model Analysis

We begin our analysis of the GAT model with a detailed review of the entropy and distribution of the attention data. This is important, because it is the attention mechanism of the GAT model that increases the performance of the model over traditional text classification models such as the baseline linear BERT classifier.

For any node i, α_{ij} forms a discrete probability distribution over all its neighbors with the entropy given by:

$$H(\alpha_{ij\in N(i)}) = -\sum_{j\in N(i)} \alpha_{ij} \log \alpha_{ij} \tag{7}$$

A low entropy means a high degree of concentration, and vice versa. For instance, an entropy of zero means all attention is on one source node. An analysis of the attention entropy across all heads for the GAT model shows a high degree of entropy, and therefore a high attention dispersion (see Figure 2). This conclusion is supported when when we look at the frequency distribution of attention weights across the edges (see Figure 3) and the graph network as a whole (see Figure 7).

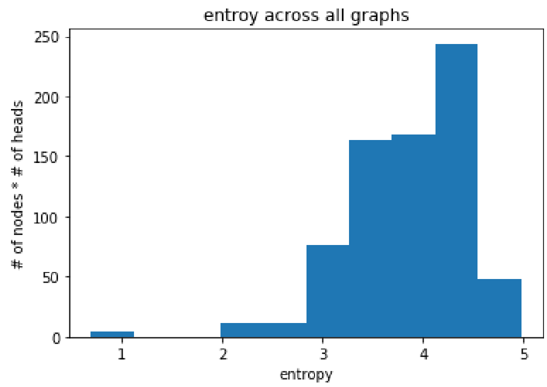


Figure 2: Entropy of Attention Weights Across all Heads

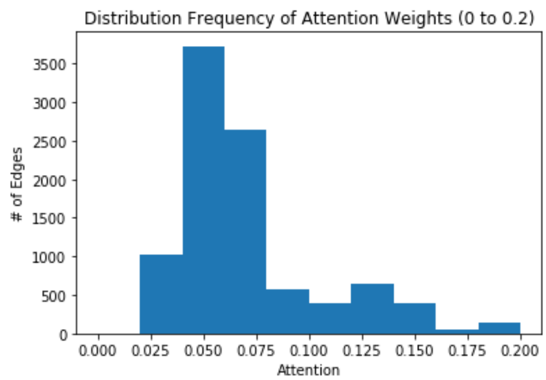


Figure 3: Distribution of Attention Weights Across Graph Edges (range 0 to 0.2)

We conclude from this analysis that most of the relationships in the GAT model have little impact on the ability of the model to predict whether a news article is fake or not, so we need to concentrate our analysis on the edges with the highest attention weights. Total edge attention was selected in favor of total node attention as, node attentions depended heavily on how many neighbors a particular node had, rather than on the attention itself (see Table 3).

Our analysis determined that the attention weights of in_edges were fairly uniform, whereas

rank	node	attention	neighbors
1	31	8.8509	145
2	71	7.1705	122
3	41	7.0325	113
4	139	7.0199	116
5	5	6.4779	116
6	86	6.1613	104
7	180	5.6515	104
8	179	5.5093	103
9	92	5.4524	93
10	84	5.4277	84

Table 3: Top 10 Nodes by Aggregate Attention

the attention weights of out_edges had some variation, so we proceeded to rank order all the out_edges of the GAT model by their aggregated attention weights across all 4 heads (see Table 4).

rank	out_edge	attention	edge list
1	2047	2.0021	(162, 62)
2	9632	1.9979	(62, 62)
3	6044	0.4464	(86, 38)
4	**399**	**0.4449**	**(71, 38)**
5	2896	0.4447	(9, 38)
6	9608	0.4461	(38, 38)
7	8835	0.4443	(52, 38)
8	8238	0.4439	(152, 38)
9	9171	0.4438	(149, 38)
10	**8887**	**0.4438**	**(89, 38)**

Table 4: Top 10 Out-Edges by Aggregate Attention

A review of the edge attention weights shows a sharp drop-off of aggregate attention weight after the first two edges. It was determined this was due to edge 2047 being the only edge shared between nodes 162 and 62, and edge 9632 being a self-referential edge to node 62. For this reason, both edges were dropped from our analysis. We selected edge 399 (node 71, node 38) and edge 8887 (node 89, node 38) for our analysis as they shared an interesting node (node 38) in common.

After plotting the 1-hop node neighborhood for nodes 89 and 71 it became clear that both nodes had an out-sized attention influence on node 38 relative to the other nodes in their network (see Figures 8 and 9). Given this clear signal, we decided to proceed with our LIME analysis on the content of the news articles embedded in these three nodes (38, 71 and 89).

4.4 LIME Analysis

We initially selected 10 features when applying LIME to the news articles for nodes 38, 71, and 89. The output are the key weighted features of a linear model which approximates the behavior of our combined BERT embedding with GAT model. We see that in every example, the locally interpretable model includes ”stop words” such as ”this”, ”the”, ”as”, ”to” and ”it”, making interpretability difficult. At the risk of lowering the local fidelity, we reduced the LIME parameter to 5 features to see if we could improve interpretability. The results were mixed, demonstrating the trade-off between interpretability and fidelity. For node 89, we eliminated the word ”a”, but also possibly informative words such as ”USAforTrump2016” and ”twitter” (figures 4 and 10). For node 38, we are still left with the uninformative words, ”this” and ”the”, however; we see that the word ”Well” is preserved which forms an interesting pattern with node 89, which also includes the same word (figures 5 and 11). One interpretation is that the use of the word ”Well” is a stylistic of a an entity known to publish fake news. Finally, for node 71, we see that all the stop words are removed, leaving words such as ”corrupt”, ”Hillary” and ”Clintons” - terms commonly used together in conspiracy theories (figures 6 and 12).

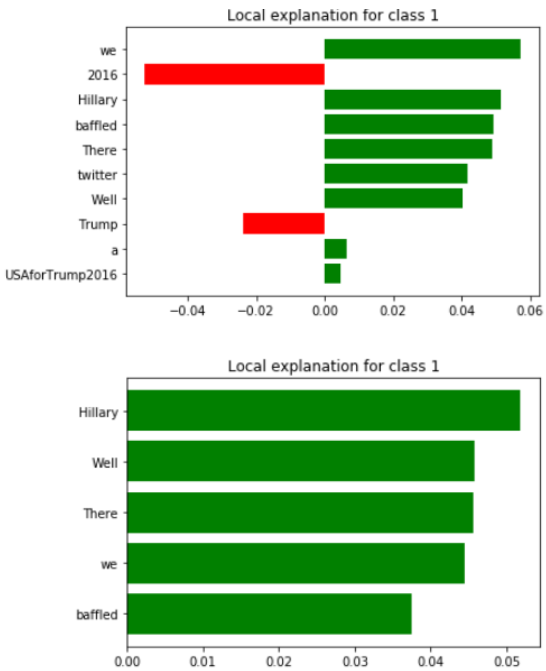


Figure 4: LIME Analysis of Node 89 Content

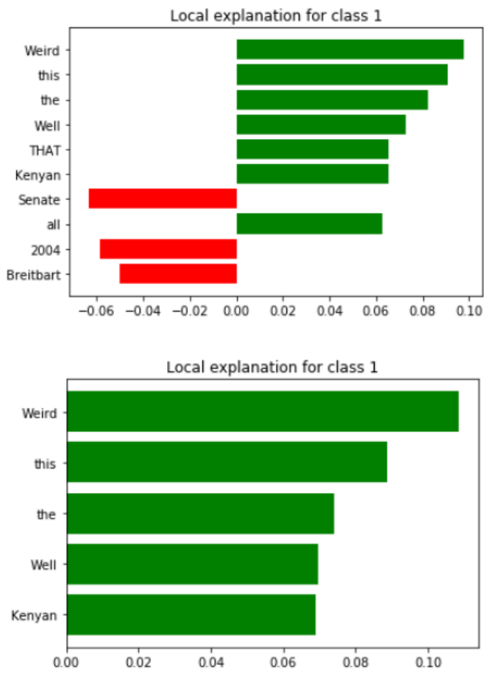


Figure 5: LIME Analysis of Node 38 Content

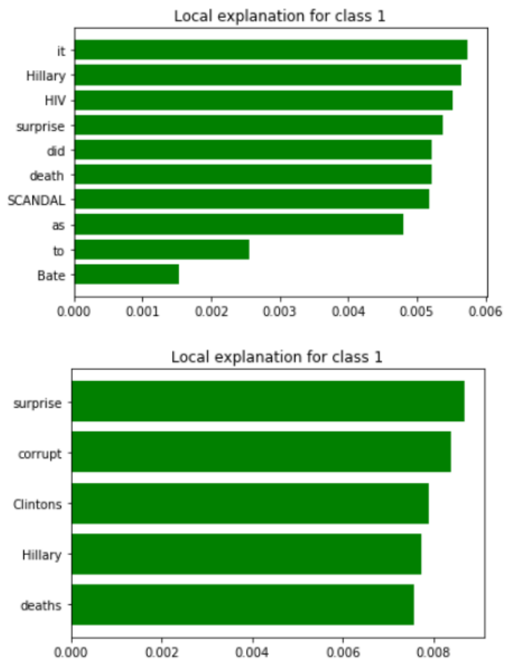


Figure 6: LIME Analysis of Node 71 Content

5 Conclusion

We introduce a new process for detecting fake news by combining BERT embeddings with GAT modeling and explaining the results with LIME outputs to identify and analyze interesting relationships between news articles. Our proposal demonstrates how to take advantage of graph relationships and add explainability to help users understand why a fake news article may be classified as fake. However, this work requires extensive data of the graph relationship between news articles, which may not always be available.

Future research could explore the use of synthesized graphs for graph regularization which use a similarly metric such as the 'L2' distance, 'cosine' distance, etc., to infer relationships between pairs of samples[5] Using such a synthesized graph approach could make this type of analysis available to a broader spectrum of use cases, to include near-real time streaming analytics to help combat the spread of fake news and misinformation as it is propagated across a social network.

Additional explainability techniques for graph convolutional networks, a nascent and rapidly developing field, should also be explored in the context of fake news detection, including guided backpropogation (GBP) and layer-wise relevance propogation (LRP)(see, for example, Baldassarre, Azizpour, 2019). SHapley Additive exPlanations (SHAP) may be a more appropriate explainability method to fake news detection, although it is more costly.[6] SHAP's approach can work more efficiently than LIME for improved accuracy and consistency when using certain models and could serve as a better representation of explainable fake news detection models.

References

Alvarez-Melis, David, and Tommi S. Jaakkola. 2018. *"On the robustness of interpretability methods."* arXiv preprint arXiv:1806.08049 (2018).

Baldassarre, Federico, and Hossein Azizpour. 2019. *"Explainability Techniques for Graph Convolutional Networks."* arXiv preprint arXiv:1905.13686

Chayapathy, Aditya; Manickam, Arun; Basavaraju, Jagdessh; Nudurupati, Anuhya; Shreesh, Abhijith 2019. *"Fake News Detection Using GATs"* `https://github.com/iamjagdeesh/Fake-News-Detection`

Cui, Limeng, et al. 2019. *"dEFEND: A System for Explainable Fake News Detection."* Proceedings of the 28th ACM International Conference on Information and Knowledge Management. ACM

[5] See, for example, "Graph regularization for sentiment classification using synthesized graphs", `https://www.tensorflow.org/neural_structured_learning/tutorials/graph_keras_lstm_imdb`

[6] https://github.com/slundberg/shap

Devlin, Jacob, et al. 2019. *"Bert: Pre-training of deep bidirectional transformers for language understanding."* arXiv preprint arXiv:1810.04805

Lundberg, Scott and Su-In Lee. 2017. *"A Unified Approach to Interpreting Model Predictions."* Advanceds in Neural Information Processing Systems 30 (NIPS 2017)

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. *"Why should I trust you?: Explaining the predictions of any classifier."* Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM

Shu, Kai, Suhang Wang, and Huan Liu. 2017 *"Exploiting tri-relationship for fake news detection."* arXiv preprint arXiv:1712.07709

Veličković, Petar, et al. 2018. *"Graph attention networks."* arXiv preprint arXiv:1710.10903 Published as a conference paper at ICLR 2018.

Yang, Shuo, et al. 2019. *"Unsupervised fake news detection on social media: A generative approach."* Proceedings of 33rd AAAI Conference on Artificial Intelligence.

Zhang, Hoa, Mufei Li, Minjie Wang Zheng Zhang 2019. `https://docs.dgl.ai/en/latest/tutorials/models/1_gnn/9_gat.html`

6 Appendix

This project relies in part and is adapted from a number of repositories available at `https://github.com/mastreips/2019-fall-main/tree/master/project/w266_Final_Project`

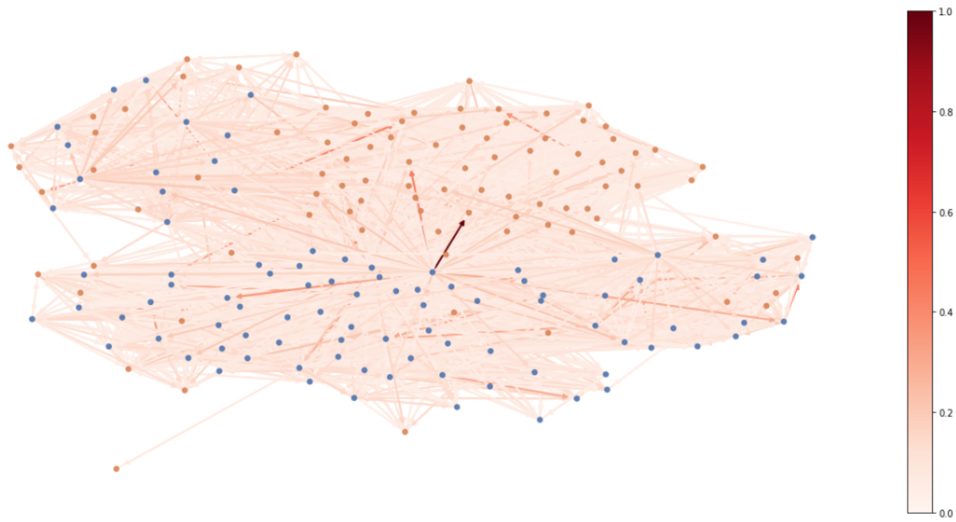


Figure 7: All nodes from GAT Analysis

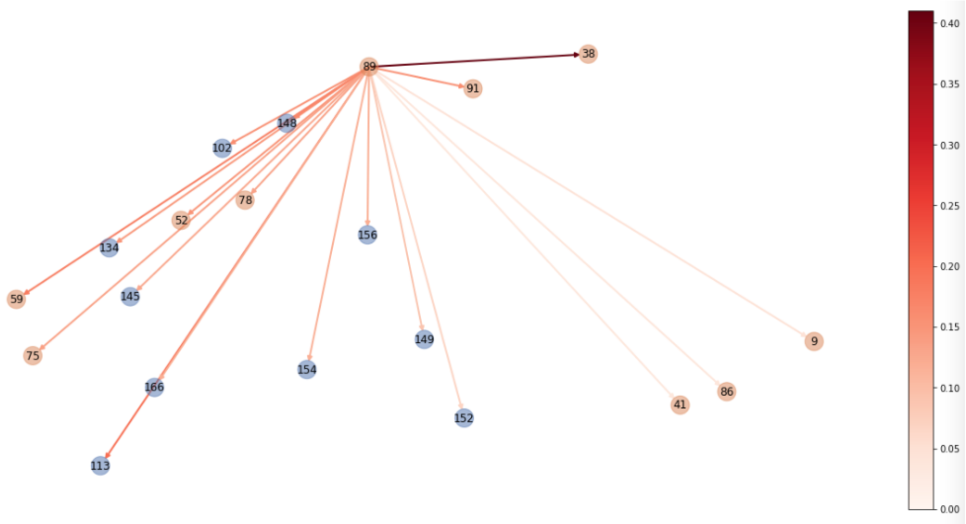


Figure 8: Node 89 from GAT Analysis

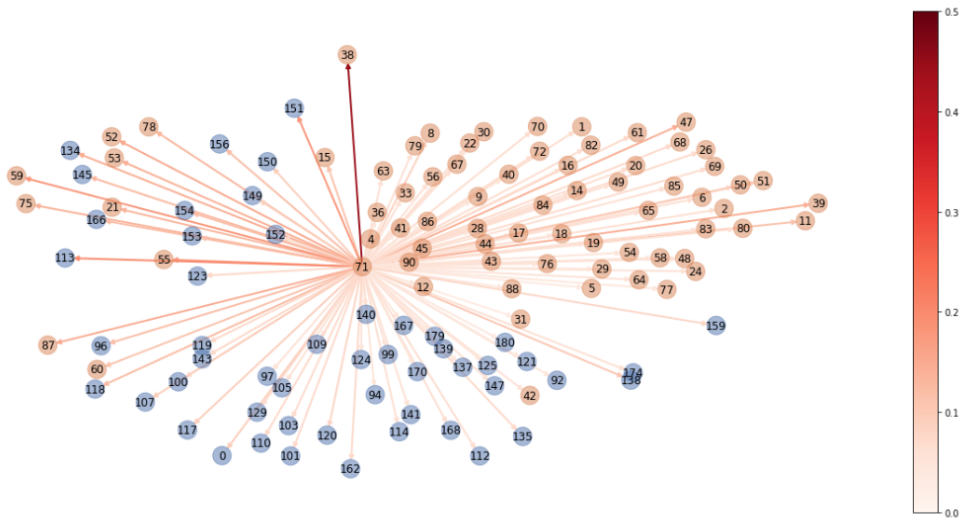


Figure 9: Node 71 from GAT Analysis

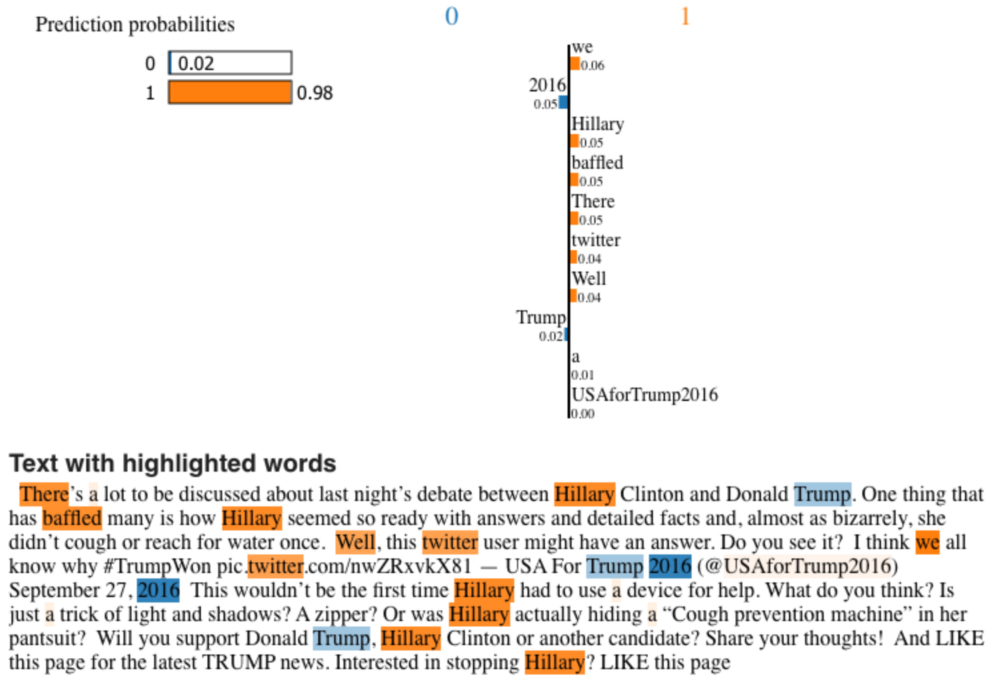


Figure 10: LIME Analysis of News Content in Node 89

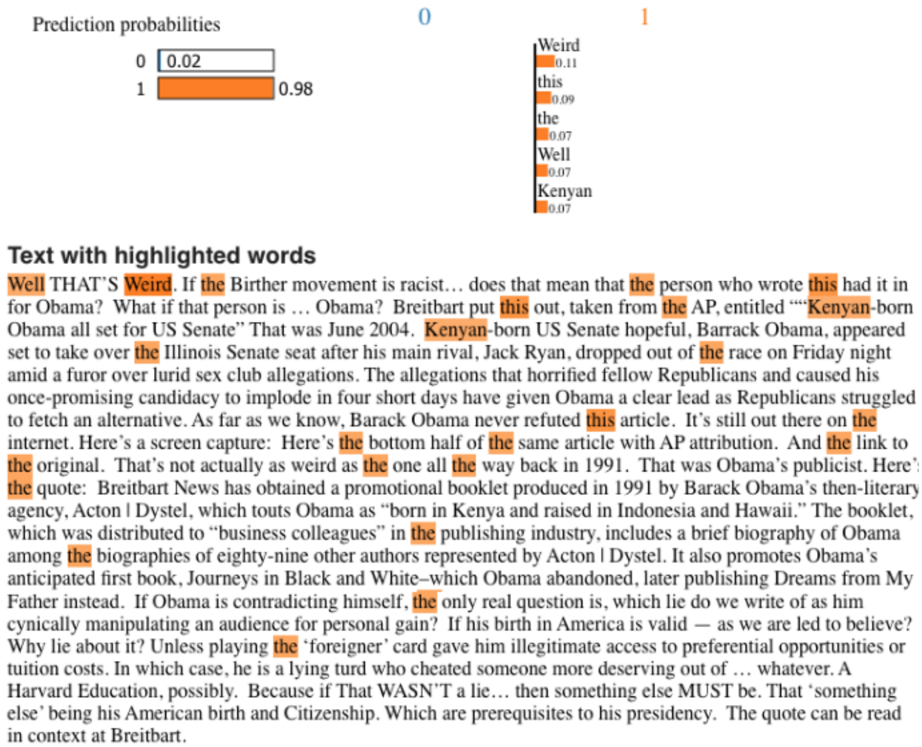
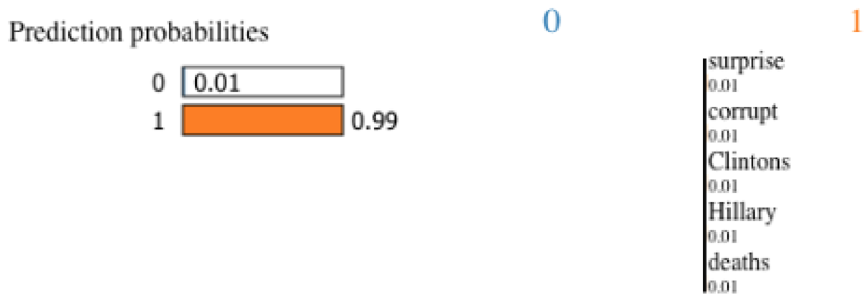


Figure 11: LIME Analysis of News Content in Node 38



Text with highlighted words

New Clinton Foundation SCANDAL Just Broke – This Is What Hillary ACTUALLY Thinks About Black People… This is just criminal and it is no surprise it involves the Clinton Health Access Initiative. Anything with 'Clinton' in it has to be corrupt, but this resulted in the deaths of many, many Africans. They distributed "watered-down" HIV/AIDs drugs to patients, virtually sentencing them to death. An Indian drug manufacturer called Ranbaxy was behind this… the question is, did the Clintons know about it. I'm betting that they did. Ranbaxy was eventually found guilty in 2013 on seven counts with intent to defraud and the introduction of adulterated drugs into interstate commerce. The DOJ also levied a $500 million fine and forfeiture against the company. So, one wonders why the Clintons were not also looked into and questioned on all of this. "This is the largest false claims case ever prosecuted in the District of Maryland, and the nation's largest financial penalty paid by a generic pharmaceutical company," said US Attorney for the District of Maryland Rod J. Rosenstein when Ranbaxy pleaded guilty. From The Daily Caller: Former President Bill Clinton and his Clinton Health Access Initiative (CHAI) distributed "watered-down" HIV/AIDs drugs to patients in sub-Saharan Africa, and "likely increased" the risks of morbidity and mortality, according to a draft congressional report obtained by The Daily Caller News Foundation. The congressional report, titled,"The Clinton Foundation and The India Success Story," was initiated by Rep. Marsha Blackburn, a Tennessee Republican and vice-chair of the House Energy and Commerce Committee. The CHAI program to help AIDS victims is considered one of the Clinton Foundation's most important contributions and is probably its best known initiative. × 101 Things All Young Adults Should Know by Sir John Hawkins John Hawkins's book 101 Things All Young Adults Should Know is filled with lessons that newly minted adults need in order to get the most out of life. Gleaned from a lifetime of trial, error, and writing it down, Hawkins provides advice everyone can benefit from in short, digestible chapters. Buy Now The congressional report focused on Clinton's decade-long relationship with a controversial Indian drug manufacturer called Ranbaxy, which CHAI used as one of its main distributors of HIV/AIDS drugs to Third World countries. It also highlighted the work of Dinesh Thakur, a former Ranbaxy employee who became a star whistleblower, permitting the U.S. government to launch a landmark lawsuit against the Indian firm. The company was vulnerable to U.S. prosecution because it also sold its generic drugs on the U.S. market. Ranbaxy ultimately pleaded guilty in 2013 to seven criminal counts with intent to defraud and the introduction of adulterated drugs into interstate commerce. "The question becomes, 'how many people lost their lives, how many people found it was a false promise,'" asked Blackburn in an interview with the DCNF. This could definitely bring further scrutiny to the Clinton charities and to the Clintons themselves. Right when they need it least. A congressional study has uncovered inappropriate ties between Bill Clinton and the two Indian-Americans who were investigated and sanctioned by the FDA and the SEC. What is very troubling is the Clinton Foundation's aggressive promotion of Ranbaxy despite mounting evidence that the Indian firm had persistently poor quality control and attempted to cover it up through either faulty or fraudulent reporting to the FDA. It is unclear at this juncture how many AIDS patients received the "watered-down" drugs. But that it happened at all, surely resulted in many deaths and is just monstrous. All for money. ProPublica estimated that in 2007 alone, the US Agency for International Development allocated $9 million to Ranbaxy and delivered "more than $1.8 million packages." "Substandard HIV medicines cause health problems for patients, perhaps even accelerating death from HIV-related infections," Roger Bate, an economist at the American Enterprise Institute who researches substandard and counterfeit medicines, told the DCNF. Ranbaxy's generic drugs are banned in the US, but they are still selling and distributing HIV/AIDs drugs worldwide. The Clinton Foundation and Clinton's charity have not severed ties with them either. Companies that partner with The Clinton Foundation and charities make a ton of money. Corruption is rampant. Bill and Hillary Clinton obviously care nothing about black people or humans in general… their primary interest is monetary.

Figure 12: LIME Analysis of News Content in Node 71