Global Explainability for understanding opinions on social media

Moritz Schüler
Department of Computer Science
Technical University Munich
moritz.schueler@tum.de

Amine Bentellis
Department of Computer Science
Technical University Munich
amine.bentellis@tum.de

Arpit Karwasara
Department of Computer Science
Technical University Munich
arpit.karwasara@tum.de

Abstract

In this age of social media we have a immense amount of data that these platforms can provide. It becomes extremely hard to find new and innovative ways to extract value from that data. In this paper we work on analyzing the impact, ideology and stance of several social media channels to enhance the understanding of opinions on social media. For this we collected our own dataset in the form of Facebook posts from CrowdTangle and also used the SemEval Stance dataset consisting of tweets with annotated sentiment and stance. Additionally we provide more insights into the model output by using several global eXplainable Artificial Intelligence (XAI) techniques. Specifically, we leverage SAGE and Neuron Attribution for global explainability.

1 Introduction

As more and more attention shifts towards the online life, the demand for safety is increasing. One area for improvement is in the amount and type of fake news. With new technologies it is very simple to construct fake images and even possible to create fake videos or speeches. However the easiest and most effective way remains plain text with wrong statements. The possible impact is far-reaching as everyone is connected and the amount of data is huge. Every minute over 340 thousand photos are posted on Instagram and over 40 million messages are sent via Whatsapp according to DOMO (2020). To handle these numbers automated processes are needed. These are challenging to develop, especially for text data. Because text per se is not a number, which can be easily interpret by some algorithm. Hence it needs to be transformed, but in a clever way, as words can have very different meanings depending on the context they are used in. By using novel techniques in the area of Natural Language Processing (NLP), we aim to analyze Facebook posts and tweets regarding their impact, ideology and stance to enhance the understanding of opinions on social media. However only achieving a high accuracy is not enough to truly get the idea, why a certain post was shared several times, whereas a similar one got no attention at all. Therefore, we complement our results with model interpretations using the field of eXplainable Artificial Intelligence (XAI). This helps us to not only identify high performing posts, but also reason about their key attributes, like special words.

In the next section we gather related work on understanding social media and explaining model outputs. Afterwards we describe the used XAI methods for this work. In section 4, we elaborate on our approach and propose a pipeline which can be used for various text related tasks. Our results are presented in section 5, where the model performance as well as its explanations are described in detail. Lastly, future research directions are outlined.

2 Related Work

In 2017 Vaswani et al. (2017) proposed the first transformer architecture, which enables short and long term context by clever connecting related words, which was a breakthrough in the NLP domain. The major downsides were the high parameter number and the slow training time. Thanks to transfer learning and BERT, a pre-trained transformer model for language modelling tasks, first introduced by Devlin et al. (2018) the drawbacks could be attenuated. Further advancements could be made by Sanh et al. (2019) which used the technique of distillation to create DistilBERT, which achieves similar performance with only a fraction of the parameters of the full BERT model. Many more techniques exist to improve the efficiency of transformer model, which could be applied to

the NLP domain. Tay et al. (2020) gives a nice overview of several promising methods.

One applicable domain for transformers is the analysis of social media and recently especially the corona crisis is a hot topic. Weinzierl and Harabagiu (2021) uses graph neural networks to understand misinformation spread. Other studies investigate fake news directly on specific Facebook pages (Boberg et al., 2020). Another approach, presented by Bhuiyan et al. (2021) tries to assess a news credibility score to hinder misinformation spread. Further ideas are to design recommender systems, specifically trained to depolarize (Stray, 2021). Others work on a more sentence based method and try to predict hate speech (Dhamija et al., 2021; Bhatia et al., 2021), sentiment (Dutta et al., 2021; Jones et al., 2021) or stance (Grimminger and Klinger, 2021; Hardalov et al., 2021), which can be used to further understand polarizing posts or find groups of similar people. Overall, the systems achieve quite good performance, however they all lack one thing, which is explanations. How did the model came to its decision.

As of now there are not really many papers available combining social media analysis and XAI. One very recent example is by Agarwal (2021), which tries to understand the farmer riots in India in January of this year. There is also a paper concerning the covid 19 pandemic, which again tries to understand the misinformation spread (Ayoub et al., 2021). In our work, we want to complement our classification with explanations as well, to reveal the key factors for success of black box models and shed light on existing problems to further investigate. Additionally, we add novelty by analyzing our model predictions on a global level using XAI.

A good entry point about existing XAI methods and their applicable domains is given by Lakkaraju et al. (2020) and Molnar (2021). In general methods can be divided in several categories. Some explain the predictions while inference, whereas others only provide a post hoc analysis. Furthermore, a difference exists in the type of explanation. Some methods provide local explanations, meaning they only explain the model output for a single specific sample, whereas global XAI methods concern the whole dataset. The most commonly used framework is SHAP, first introduced by Lundberg and Lee (2017). It stems from game theory and analyzes the importance of each individual feature for a specific sample. Looking at a single image makes it a local XAI method. Covert et al. (2020) on the other hand generalized the idea to a global setting, where one gets the individual feature importance over all samples. A different way for analyzing model predictions is to look at every neuron. Again, locally extracting which neurons were responsible for the model output or in a global manner by simply averaging the number of activations of every neuron across the data. One method of this kind was developed by Dhamdhere et al. (2018). A combination of the described method can be used to gain a good understanding about a model's predictions, however the outcome of each method is very abstract and artificial. Therefore, Kim et al. (2017) came up with the idea of "Testing with Concept Activation Vectors" (TCAV). In this recent work, the authors describe concepts by specifying several labeled data points, that adhere to a concept and the same amount, which do not. With this information the TCAV method can quantify the sensitivity of the underlying model to the concept under review. In the next section all used XAI methods for this work are described in more detail to give the reader a intuitive understanding to easily follow the results in section 5.

3 XAI Methods

The interpretation of deep-learning models can be challenging due to their *black-box* like nature. It is especially applicable for NLP models, where the nuances of writing and big vocabularies also make the task more complex. In this section we will give a short introduction to the XAI methods we used for our project. First, we will present SHAP, then we will try to give an intuitive description of SAGE (Shapley Additive Global importancE). It is necessary to understand SHAP first because it is SAGE's building block.

3.1 SHAP

Local explanability are methods aiming at explaining individual samples, it is similar to asking *Why does this specific post adheres to this ideology?* SHAP is a common local explanation method using Shapley values which is a game theory concept (Lundberg and Lee, 2017).

Let's consider **f**, our model, taking in features $\mathbf{x} = (x_1, x_2, ..., x_d)$ and outputs **f(x)** the prediction. What SHAP tries to accomplish is find how each feature x_i influenced the prediction. In the political context, which key-words or sentences

makes the model think that this particular post is Democrat (or Republican). More precisely, what is the value $\phi_1, \phi_2, ..., \phi_d$ assigned to each feature $x_1, x_2, ..., x_d$, essentially making a map of the contributions of all features used for the prediction. The values Φ are computed by only providing a subset of the features **x** to see how depriving certain information will move the prediction, positively or negatively. We will now explain how SHAP and SAGE are linked.

3.2 SAGE

One major difference between SHAP and SAGE is that the latter tries to understand the effect of $x_1, x_2, ..., x_d$ overall (local vs global). So we no longer try to explain a single sample but the whole dataset by averaging the values we get for each $(\mathbf{x}, \mathbf{f(x)})$ pair in our dataset. It also uses a different version of SHAP named LossSHAP. Indeed, instead of thinking about how each feature changes the prediction, we want to know how much each feature affects the prediction's accuracy. In other terms, features that make the prediction more accurate have large values $\phi > 0$ and features that make the prediction less accurate have negative values $\phi < 0$ (Covert et al., 2020). The results we get after averaging the LossSHAP values across all the dataset is the contribution of each feature to the model's performance, and they are named SAGE values.

When applying SAGE to NLP problems, each word of the vocabulary will be considered as a feature. This is a problem because there are thousands of words in our vocabulary which means that the SAGE values will be really sparse and thus not interpretable. One way of solving this problem is to cluster words with similar meaning (credits to the **SHAP-team** for this method). Instead of feeding SAGE the whole vocabulary, we provided around 10 to 20 groups as features. For example, one group common to almost all the experiments we performed was the "Names" group (i.e. [kelly, joe, glenn, atkinson, jon, chuck]). We further added "Punctuation", "Stop-words", "Syllables" and "Digits" as other features.

3.3 Neuron Attribution

Neuron Attribution is basically quantifying the importance of a neuron for an given set of input values with respect to a particular class. For example in the classification task "Democrat" vs "Republican" for a set of text values, we can determine the Neuron Attribution values for prediction the "Democrat" class using attribution techniques. There are several approaches to calculate Neuron Attribution.In our case we used Layer Attribution, which is basically Neuron Attribution value for every Neuron in a specified layer. We calculated Layer Attribution for Linear tail layers in our model.

One of the techniques used for Neuron Attribution is Neuron Activation. It is a very simple technique and it simply returns the activation value of a neuron for a given input. We can calculate then the average activation value over a set of inputs. Another technique for Neuron Attribution is Neuron conductance, which is described in this paper (Dhamdhere et al., 2018). Conductance is used to understand the importance of a hidden unit for the prediction of a set of inputs. Conductance uses Neuron Activation and the partial derivatives of the neuron with respect to the input plus with that of the output with respect to the neuron to build a more complete picture. Conductance through a neuron can be understood as the flow of attribution through a neuron.

4 Our Approach

In this section we propose a general pipeline for text based classification tasks. Additionally, we present the specific tasks for our work and describe the datasets used for them.

4.1 Pipeline

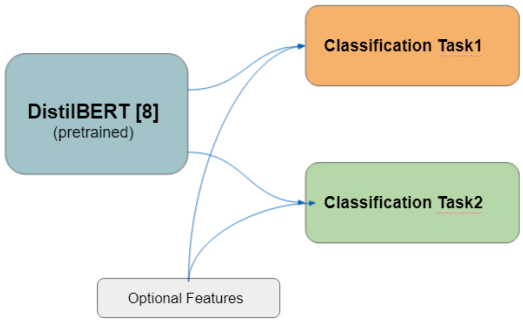

Figure 1: Our Modelling Pipeline

Here we present our general pipeline for our tasks and our dataset overview. We perform classification tasks on our datasets using this pipeline. As mentioned earlier we are using DistilBERT in our pipeline. DistilBERT is basically a smaller, faster, cheaper and lighter version of it's predecessor BERT model. We can fine tune DistilBERT

further for any specific NLP task, that is why we use it in our Pipeline and further optimize it for specific tasks by adding additional layers on top of it. Further we also tried several approaches to explain our models at a global level. These approaches were described earlier in the previous section. Our eXplainable AI methods are also flexible in a sense that they can be used with all of our use cases. So our pipeline can be considered as a base model and can be easily extended for different use cases.

4.2 Applications

We perform two main classification tasks, one is stance prediction for a tweet and the other one is party + impact prediction for a Facebook post. For both tasks the above described pipeline is used. The only difference is in the last layers, which differ in the number of output neurons dependent on the classification task at hand. For the stance classifier two classifications are made. One is to classify whether a sentence is neutral, in favor or against a specific topic and the second classification tries to predict the topic the sentence adheres to. As a loss function standard cross entropy loss is used and the optimizer is Adam. Best results (presented in section 5.2) could be achieved with a learning rate of 0.0014, a batch size of 32 and training for 20 epochs with early stopping enabled. The optimization metric chosen was the F1-score.

For the party+impact prediction our model consists of a pre-trained DistilBERT model attached to two Linear Layers which are called Party Layer and Score Layer. These two layers are trained separately from each other and have separate weights. The result from Party Layer is fed to a Binary Cross Entropy loss function which is further used in backpropagation on Party layer. Similarly, the result of Score Layer is fed into a Binary Cross Entropy function and the result is used in it's backpropagation step. We are giving the Post Text, Post Type and Post Date as input features to our model. In this task we try to predict the Party to which an facebook Post belongs to, which is either Republican or Democrat. Also we predict the impact a post has, either "Popular" or "Non-Popular".

In the following subsection the two used datasets for the above described applications are outlined in detail.

4.3 Datasets

For the first application of stance classification we used the great dataset from Mohammad et al. (2016), where the authors labeled around five thousand tweets with a sentiment and stance. Furthermore, the data covers six different topics, e.g. climate change. With the dataset Mohammad et al. (2016) also provides a nice interactive data visualization tool (Mohammad, 2016), where one can gain first insights about the data distribution.

To predict the ideology, in our case the party affiliation, and the impact of a post, we collected posts from several Facebook pages belonging to two US political parties, Republicans and Democrats. We wanted the dataset to be easily reproducible, so we collected posts from official Party handles of the politicians and regional party branches. This gave us a dataset of five thousand posts for each party. We built a classifier to label each post as either Republican post or Democrat. We also used a score metric for each post which predicts a post's popularity. Another of our goal was to perform regression on the score but we were not able to get good enough results. So, we converted the regression task for score into a classification task, in which we basically predict if a post is popular or not. For this task we added "popular" label for every post with a score higher than 1 and "not-popular" label otherwise.

5 Results

In this section we will discuss our results, more precisely, the global explanations we are able to produce. Two different methods were used: SAGE on both the stance and CrowdTangle dataset, and Neuron Attribution on the latter one.

5.1 Stance Classification

With the pipeline design presented in section 4.1, the trained model for stance classification is able to achieve a F1-score on the official test dataset of 0.65. The creator of the dataset, Mohammad et al. (2016), also launched a challenge for stance prediction in 2016, where the winning model achieved slightly higher values of 0.67. Although having a lower performance score, the robustness and generality of our model could be higher. To check, how well the model generalizes we examined SAGE values to find important word groups and checked manually, if those word groups are coherent and make intuitive sense for stance prediction. In the following image, the SAGE values for a clustering into 17 bins are displayed.

Overall, one can see that there is only one group

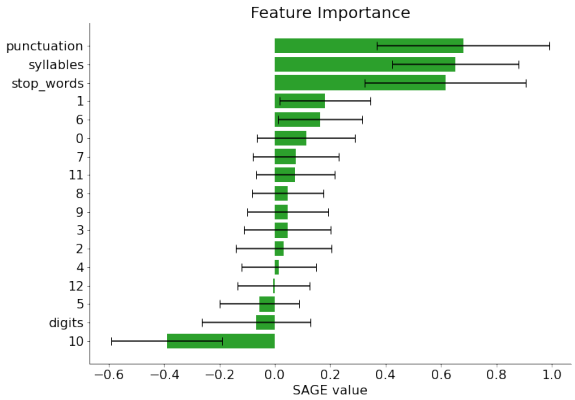

Figure 2: SAGE values for stance prediction using 17 clusters.

of words with negative values, meaning that these words lower the prediction accuracy, whereas all positive groups help the model to predict the stance accurately. All groups with a value around zero, don't have any impact on the prediction of the model and could theoretically be removed. But in our case of text data, where the features are word groupings, removing is not possible because the BERT model depends on the context words are placed in. This means removing these words would drastically change the prediction of the model.
One example for a group of words with no impact is group eight. Examples clustered in there are names, like "Netflix" or "Volvo" and neutral words like "minute" or "second". In the only significant negative group, we get a lot of randomly put together words, which could mean that the clustering was able to find a noise group, that doesn't exhibits a specific pattern. Another interesting remark is that the most important features are punctuation, syllables and stop words. The first grouping makes intuitive sense because putting an exclamation mark at the end of a sentence often expresses some kind of stance. However the other two word clusters don't make sense to a human. The amount of syllables and the usage of stop words usually have no relation with the stance. Therefore, one can see that although the performance measured with the F-score is quite good, the model does not seem to generalize very well to this task.

5.2 Party Classification

As mentioned in the *Section 4*, our model consists of two linear layers, one for Party classification and one for Score classification. First, we need to make sure that the accuracies of our models are good before attempting any explanations on them. So, we managed to achieve an accuracy of around 0.78 and 0.71 for both Party and Score respectively. However, the explanations results presented in this section focus on the Party tail. The results on the Score tail are not satisfactory enough for us to include them here.

Using the clustering approach discussed in the *3.2 Section*, we are grouping the vocabulary to use SAGE on our social media dataset. Indeed, we find interesting insights on the groups and SAGE explanations. Positive values seem to represent democratic ideas or concepts and negative values are more republican-based.

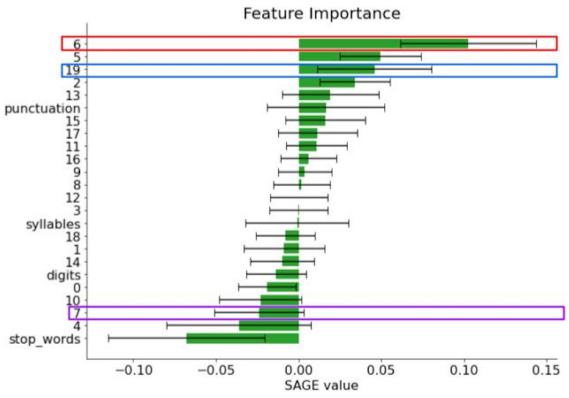

Figure 3: SAGE values using 20 clusters and 50 dataset samples. Each number represents a specific group (features). Red group represents the "Education" concept, blue is the "State/Government" concept and purple is the "Republican" concept.

Moreover, stop-words often have the lowest SAGE value 3, and it looks as though the number of words inside the group influences the feature importance value. Not all the results can be interpreted and it is due to the fact that the general meaning of a group is not always obvious. For example, it is possible to interpret the results of the "Names" or "Stop-words" groups because we can put names on those groups of words. In some instances the groups have no real concept or idea behind them and they look like a random set of words. One other drawback of SAGE in this context is that the interpretation is not human-friendly: it is quite hard to extract knowledge out of these values. It would probably require multiple experiments with different data sources to draw conclusions about republican or democrat speech. And it would probably be way more complex to gather insights for the *impact* predictions.

5.3 Neuron Attribution

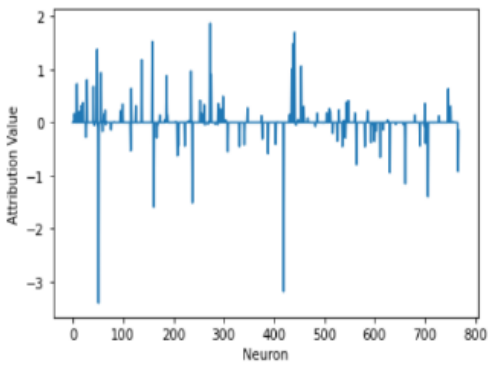

Figure 4: Neuron Attribution Results

After applying Layer Conductance to our we got Neuron Attribution value for every neuron. We used Layer Conductance only for our Party Classification model. To get Conductance for our dataset we took average of Conductance values over the Test dataset. Here we can see in 4 the attributions for Democrat class over our datset. We see that some neurons have high overall impact and ignoring that neuron during the prediction phase should shift most of the results to the opposite class. Also we can notice that most of neurons have low Conductance values, which implies that some neurons have most of the predicting power when it comes to this particular classification task.

6 Future Work

There are several promising avenues for future works as our project is only a demonstration of what can be done with XAI methods on NLP.

We can firstly talk about one of our goals when starting this project, predicting and explaining the impact of posts. Even though our *Score* classifier works properly, it was particularly hard to get explanations for those predictions. The ultimate goal would be to find patterns in social media post (inside the text and meta-data) that determines if a post will under- or overperform. And as we realized those patterns are really not obvious. It would require a lot more work and testing to abstract the essence of what makes an impactful post.

As we mentioned in the *5.1 Section* the clustering method we used did not yield the best results. Some clusters do not represent any concept or idea but instead, seem like a collection of words with no real relation between them. Also, the size of clusters was not the same across all of them. Future works can still significantly improve the explanations by switching the clustering method to have more accurate and balanced (in terms of size) grouping.

Another possibility to obtain better explanations is to use TCAV in conjunction with SAGE. TCAV is another global interpretability method that shows the importance of high level concepts (e.g., color, gender ...) for a prediction class. Our clusters can be regarded as TCAV concepts. SAGE then works as a pre-explanation step where we can select interesting concepts for our task. We would then be able to investigate in more detail those groups and maybe handcraft some groups to find the most important set of words for each class.

7 Conclusion

This project tries to answer questions like *What does make a certain post belong to an ideology group?* Or *In what ways a post can generate more or less impact?* We will conclude this paper by addressing our achievements as well as the challenges we came across. And lastly we have a small discussion on the repercussion of a hypothetical version of our project where all the questions above can be easily answered.

We faced many challenges when building this project. Our approach required gathering the data which was a challenge, in and of itself. The CrowdTangle platform helped a lot speed up this data collection phase. In addition, global explanability in NLP is a new topic and it is quite hard finding resources on this specific domain. So we had to adapt our approach to apply methods meant for other types of models.

Also, we are able to achieve decent accuracies for the *Party* and *Score* predictions (78% and 71% respectively). Even on the stance dataset where we are able to reach 65% F1-score which close to the winning model for this competition. We also managed to gather knowledge from our model using simple global explanations.

Finally, one might wonder about the ethical implications of such explanations. What will happen when we are able to guess the ideology of a person just by looking at their post or even finding a ”recipe” for impactful posts. We can imagine a future where machines work as consultants, guiding our speech to produce more reactions and to gain followers.

References

Ajay Agarwal. 2021. Tracking peaceful tractors on social media - xai-enabled analysis of red fort riots 2021. *CoRR*, abs/2104.13352.

Jackie Ayoub, X. Jessie Yang, and Feng Zhou. 2021. Combat COVID-19 infodemic using explainable natural language processing models. *CoRR*, abs/2103.00747.

Bhumika Bhatia, Anuj Verma, Anjum, and Rahul Katarya. 2021. Analysing cyberbullying using natural language processing by understanding jargon in social media. abs/2107.08902.

Md Momen Bhuiyan, Michael Horning, Sang Won Lee, and Tanushree Mitra. 2021. Nudgecred: Supporting news credibility assessment on social media through nudges. abs/2108.01536.

Svenja Boberg, Thorsten Quandt, Tim Schatto-Eckrodt, and Lena Frischlich. 2020. Pandemic populism: Facebook pages of alternative news media and the corona crisis - A computational content analysis. *CoRR*, abs/2004.02566.

Ian Covert, Scott Lundberg, and Su-In Lee. 2020. Understanding global feature contributions through additive importance measures. *CoRR*, abs/2004.00668.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. abs/1810.04805.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. 2018. How important is a neuron? *CoRR*, abs/1805.12233.

Tashvik Dhamija, Anjum, and Rahul Katarya. 2021. Comparative analysis of machine learning and deep learning algorithms for detection of online hate speech. abs/2108.01063.

DOMO. 2020. Data never sleeps 8.0.

Suchandra Dutta, Dhrubasish Sarkar, Sohom Roy, Dipak K. Kole, and Premananda Jana. 2021. A study on herd behavior using sentiment analysis in online social network. abs/2108.01728.

Lara Grimminger and Roman Klinger. 2021. Hate towards the political opponent: A twitter corpus study of the 2020 US elections on the basis of offensive speech and stance detection. *CoRR*, abs/2103.01664.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2021. Cross-domain label-adaptive stance detection. *CoRR*, abs/2104.07467.

Keenan Jones, Jason R. C. Nurse, and Shujun Li. 2021. Out of the shadows: Analyzing anonymous' twitter resurgence during the 2020 black lives matter protests. *CoRR*, abs/2107.10554.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. 2017. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). abs/1711.11279.

Hima Lakkaraju, Julius Adebayo, and Sameer Singh. 2020. Explaining machine learning predictions: State-of-the-art, challenges, and opportunities. *NeurIPS*.

Scott Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *CoRR*, abs/1705.07874.

Saif M. Mohammad. 2016. The semeval-2016 stance dataset.

Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. S16-1003.pdf.

Christoph Molnar. 2021. Interpretable machine learning - a guide for making black box models explainable.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. abs/1910.01108.

Jonathan Stray. 2021. Designing recommender systems to depolarize. abs/2107.04953.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2020. Efficient transformers: A survey. abs/2009.06732.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. abs/1706.03762.

Maxwell A. Weinzierl and Sanda M. Harabagiu. 2021. Automatic detection of covid-19 vaccine misinformation with graph link prediction. abs/2108.02314.