World Academy of Science, Engineering and Technology
International Journal of Cognitive and Language Sciences
Vol:15, No:5, 2021

Improving Subjective Bias Detection Using Bidirectional Encoder Representations from Transformers and Bidirectional Long Short-Term Memory

Ebipatei Victoria Tunyan, T. A. Cao, Cheol Young Ock

***Abstract*—**Detecting subjectively biased statements is a vital task. This is because this kind of bias, when present in the text or other forms of information dissemination media such as news, social media, scientific texts, and encyclopedias, can weaken trust in the information and stir conflicts amongst consumers. Subjective bias detection is also critical for many Natural Language Processing (NLP) tasks like sentiment analysis, opinion identification, and bias neutralization. Having a system that can adequately detect subjectivity in text will boost research in the above-mentioned areas significantly. It can also come in handy for platforms like Wikipedia, where the use of neutral language is of importance. The goal of this work is to identify the subjectively biased language in text on a sentence level. With machine learning, we can solve complex AI problems, making it a good fit for the problem of subjective bias detection. A key step in this approach is to train a classifier based on BERT (Bidirectional Encoder Representations from Transformers) as upstream model. BERT by itself can be used as a classifier; however, in this study, we use BERT as data preprocessor as well as an embedding generator for a Bi-LSTM (Bidirectional Long Short-Term Memory) network incorporated with attention mechanism. This approach produces a deeper and better classifier. We evaluate the effectiveness of our model using the Wiki Neutrality Corpus (WNC), which was compiled from Wikipedia edits that removed various biased instances from sentences as a benchmark dataset, with which we also compare our model to existing approaches. Experimental analysis indicates an improved performance, as our model achieved state-of-the-art accuracy in detecting subjective bias. This study focuses on the English language, but the model can be fine-tuned to accommodate other languages.

***Keywords*—**Subjective bias detection, machine learning, BERT–BiLSTM–Attention, text classification, natural language processing.

I. Introduction

THE presence of subjective bias in information content is a big challenge in all types of media, especially the news media. Many find it difficult trusting the content of the news as objective [1]. As reported by Gallup news [2], a number of adult Americans believe that 62% of the traditional news media and 80% of social media news is not objective. These high figures drive the need for a solution. Subjective bias can be defined as stating personal feelings or opinion as fact. Similarly, expressing general fact can be seen as objective. For example [3], the statement "Jack Ma disappears from African TV Show fueling whereabouts questions" is biased because the phrase *disappears from* presents the writers personal opinion (perceived reality) in the sentence, rather than reality itself whereas this statement "Botswana-China Talks to strengthen bilateral relations and cooperation between the two nations" is neutral. This is because it does not contain any person opinions of the writer and can be confirmed as fact that indeed such a meeting was held. To classify these, one large dataset was used in our experiments across all models: The WNC. WNC is a dataset that was put together by [4] in accordance with the Wikipedia's neutral point of view (NPOV) policy [5]. This policy is one of three Wikipedia policies that guide Wikipedia writers. The dataset consists of 360,000 sentences by English Wikipedia editors crawled from 423,823 Wikipedia revisions over a 15-year period. Analyzing the dataset, we observed that subjective bias is more prevalent in areas such as politics, sports, geography, and history than others. We further observed that contained in the biased sentences of the dataset are three kinds of bias that often appear in text: framing, demographic, and epistemological bias. By definition, framing bias are one-sided statement projecting a particular point of view (the same facts framed in different ways leading to different conclusions). Epistemological bias are subjective intensifiers that impact the believability of a statement. And lastly, demographic bias refers to bias or prejudice towards a specific demographic (members of a population), which can include sex, race, age, etc.

The main contribution of this work is the integration of the more recent BERT embedding vectors as embedding of choice for the Bi-LSTM model, rather than existing embeddings such as word2vec [6] and glove [7]. This approach shows an increase in the performance of the overall classification model. Section III expands on the methodology.

The remainder of this paper is organized as follows, the next section describes reviewed related works on the subject matter, followed by the proposed methodology in Section III, then we present the experiments on the proposed model and baselines in Section IV. And finally, Section V concludes the paper.

Ebipatei Victoria Tunyan and Cheol Young Ock are with the Department of IT Convergence, University of Ulsan, Ulsan 44610 Korea. (e-mail: ebitunyan@gmail.com, okcy@ulsan.ac.kr).

T. A. Cao is with the Department of Electrical Engineering, University of Ulsan, Ulsan 44610 Korea (e-mail: tacao@mta.edu.vn).

II. Related Works

Considerable work has been carried out on identifying subjectivity using several text classification techniques. It is no wonder, as text classification is a fundamental aspect of NLP. While some of these techniques are based on statistical methods, a majority use machine learning methods for text classification. And some (such as the proposed model) go a step further with deep learning methods. The task presented is a binary classification task that should give either a subjective or objective class as output. Recasens et al. [8] use a logistic regression-based model with linguistic feature to achieve this. They designed their model to detect the bias-inducing words in a statement. This method follows the NPOV policy which the authors introduced for this task. Pryzant et al. [4] follow a similar pattern as it builds upon the approach of [8]. Dadu et al. [9] propose the use of contextualized word embeddings to achieve this task on a sentence level. They conclude that an ensemble of these contextualized embedding models produces a higher accuracy as opposed their single counterparts. Although this comes as an improvement on the single word detection method of [4] and [8], the accuracy they present, we believe can be improved upon. And we have gone ahead to achieve that improvement in this work.

Several other models exist for the task of detecting subjectivity in text data. The authors in [10] investigated major semi supervised learning methods for identifying opinionated sentences. Riloff and Wiebe [11] focused on bootstrapping algorithms for sentence level subjectivity detection. They argue that since the subjective and objective expression patterns are based on syntactic structures, they provide more flexibility than single words or n-grams. Furthermore, they propose a dataset called the MPQA Opinion Corpus, which is a dataset containing about 5,000 subjective and 5,000 objective sentences. Compared to the dataset used by our proposed model, this dataset is much smaller. Authors in [12]-[14] proposed models that use other word embedding methods such as Word2vec, Glove, and fastText to get vector representations of their input data. In our work, we utilized the BERT [15] contextualized embeddings to generate our vector representations. This approach produces better performance as it takes into consideration the context of the input sentences. And [16] and [17] present classification models based on BERT – Bi-LSTM. We extend this by further adding an attention mechanism to capture the importance of the words in the sentence.

III. Proposed Method

For the task of subjectivity detection, we propose a deep neural network comprising of three components, a BERT model [15], a Bi-LSTM model [18], and an attention mechanism [19] (i.e., BERT + Bi-LSTM + Attention). This section gives detailed description of each component, and how they integrate to achieve the proposed model. Note that this approach adopts BERT as the upstream of the model and Bi-LSTM with attention as downstream of the model.

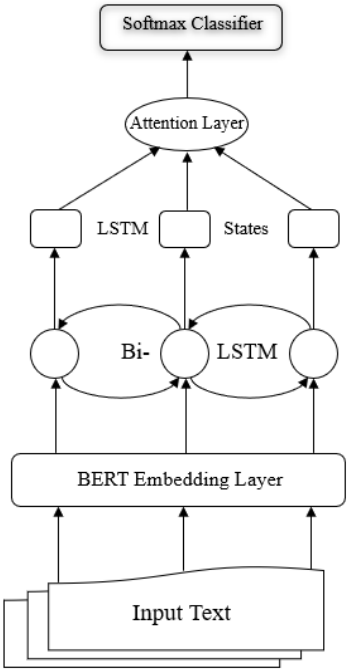

Fig. 1 Proposed model

A. BERT

Proposed by Devlin et al. [15], BERT is a pretrained contextualized text representation model that internally utilizes the bidirectional transformer network structure. Its bidirectional nature allows it to consider both directions of words in a sentence, for context. It was pretrained on a massive corpus of over 3 billion words. BERT has notable advantages over previous methods, making it more suitable for this task. As a result of its contextual nature, BERT performs well in detecting the meaning of a language sequence depending on context. This advantage enables it to recognize subtle differences in phrasing [15]. A remarkable feature of BERT is that merely using the BERT model and fine-tuning it can generate relatively good results, although building upon it gives even better performance. Results from both instances are presented in Section IV. Another key advantage is that BERT requires significantly lesser preprocessing of data compared to existing methods. Hence, BERT is adopted to preprocess all data and generate word embeddings (sequence of word vectors). We use the BERT base model as it is smaller and requires less processing time.

Given a sentence sequence $\mathrm{X} = [x_1, x_2, \cdots, x_n]$, the first step after the model accepts this sequence is preprocessing. This is done by BERT's tokenizer. It tokenizes the input sequence and maps each token into a unique ID. These token IDs serve as the input to the BERT model in the second phase of our model architecture. The output of the of the BERT layer are vector embeddings of 768 dimensions for each token of the given length.

$$\mathrm{Z} = [z_1, z_2, \cdots, z_n] \tag{1}$$

where Z is the sequence of high-quality contextualized

embeddings.

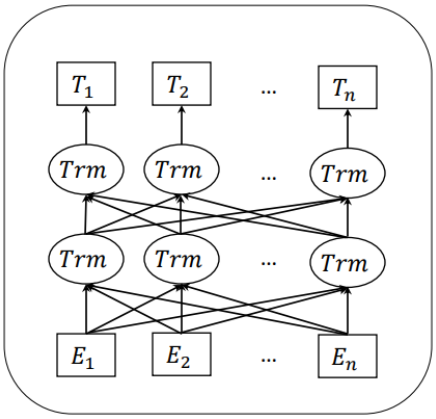

Fig. 2 BERT Structure

B. Bi-LSTM

In NLP, dealing with majority of text data require sequential processing, and regular feedforward neural networks do not handle sequences properly. Recurrent neural networks handle (variable length) sequences more accurately, as they are able to connect previous information to the present task. And the bidirectional variation such as Bi-LSTM is even better suited for such tasks, due to LSTMs [20] remarkable handling of long-term dependencies. With this advantage, BiLSTMs have achieved success in machine translation, speech recognition, and other machine learning tasks. At the core of LSTM is a gated mechanism that controls the flow of data by selectively passing information across individual time steps. BiLSTM can be understood as two separate LSTMs processing sequences forward and backward, and hidden layers at each time step are concatenated to form the cell output [21].

The Bi-LSTM layer takes as input the output of the BERT encoder to create hidden state h_t at each time step, which acts as its memory of the input sequence. Following its bidirectional nature, h_t will be updated from both the forward and the backward direction. The forward LSTM layer denoted as $\overrightarrow{\text{LSTM}}$ reads the sentence Z from z_1 to z_n and the backward LSTM layer denoted as $\overleftarrow{\text{LSTM}}$ processes the sentence in the reverse direction. The resulting hidden state h_t is a concatenation of the forward hidden state $\overrightarrow{h_t}$ and the backward hidden state $\overleftarrow{h_t}$ as shown in (2). Subsequently, the final hidden state h_t was computed by using (3); here, we applied a tanh activation function over it, parameterized by bias weight b_i and learned parameter W_i.

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right], \quad t \in [1, T] \tag{2}$$

$$h_t = tanh(W_i h_t + b_i) \tag{3}$$

Bi-LSTM initialized with the Glove word embeddings as embedding weights is capable of performing text classification, but as we would see in the subsequent section it gives a significantly less performance than that which is initialized using BERT.

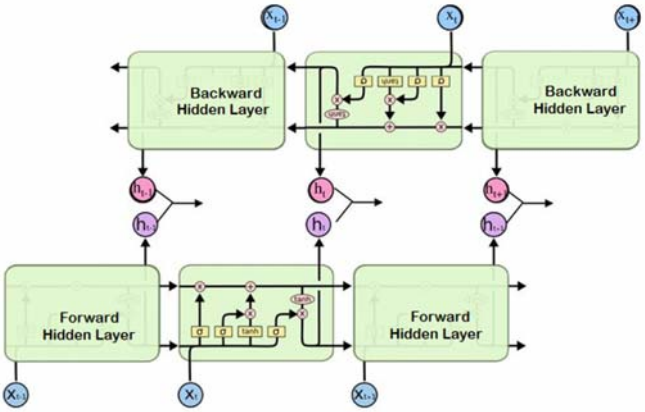

Fig. 3 Bi-LSTM Architecture [21]

C. Attention Mechanism

Proposed by Bahdanau et al. [19], attention is a mechanism that enables the network to learn to attend to information at different positions in the sequence of inputs during processing. This is especially important because not all words contribute equally (as some words may be more informative than others in constituting) to the meaning of a sentence. Attention mechanism has shown notable significance in sequence processing, hence its application in this task. The system uses the attention mechanism to capture distinct information from the context words.

Following the network flow, the output h_t of the previous layer becomes the input of the attention layer. The resultant output of this layer is subsequently the normalization of the correlation between the final hidden state h_t and a randomly initialized (learned) context vector. The hidden state is then weighted and aggregated based on the attention weight vector to generate the high-level sentence vector representation.

$$H = \sum_{t=1} h_t \alpha_t \tag{4}$$

where α_t is the attention weight matrix that contains different degree of weights for corresponding words in a sentence. In using attention mechanism, we can derive visual insights on which tokens the model learns to focus on or attend to in each sentence. A visual attention can be seen in Section IV.

D. BERT-BiLSTM-Attention Model

Finally, using (5), sentence vector H —having passed through the full BERT-BiLSTM-Attention model— is fed to a single-layer fully connected softmax classifier (which is a normalized logic function) to obtain the predicted probability distributions of classes i.e., the label for each sentence.

$$V = softmax(w_k H + b_k) \tag{5}$$

where V is the output of the model. w_k and b_k are learned parameters of the classification layer. As the network trains, it aims to minimize loss, using the cross-entropy loss function.

To summarize the method and full system training process depicted in Fig. 1, first we carry out preliminary preprocessing of the input data to clean the data and also get rid of NaN values

(i.e., rows with empty text) that may appear in the dataset. Then we preprocess the cleaned dataset using BERT to tokenize and obtain the contextualized word embeddings after which the embeddings are sent to the BiLSTM layer for processing. And attention is subsequently applied to it before it reaches the softmax layer that classifies and outputs the result.

IV. Experiments

We evaluate the performance of our model in this section. It covers all experiments carried out, experimental setup, model implementation, and lastly, results and discussion. But first we describe the dataset used for the experiment.

A. Dataset

For experimental purposes, we utilized the WNC dataset which consists of 180,000 subjectively biased sentences and an equal 180,000 neutral sentences. Having an equal proportion of both classes make for a balanced dataset, as shown in Fig. 4. We can further see in Table I the proportion (in percentage) of the kinds of bias present in the biased dataset. We divided the dataset into training set, validation set, and test set, at 70%, 20%, and 10% respectively.

TABLE I
Proportion of Bias Subcategories in the Biased Sentences [3]

Sr. No	Subcategory	Percent
1	Framing	57.7
2	Demographic	11.7
3	Epistemological	25.0

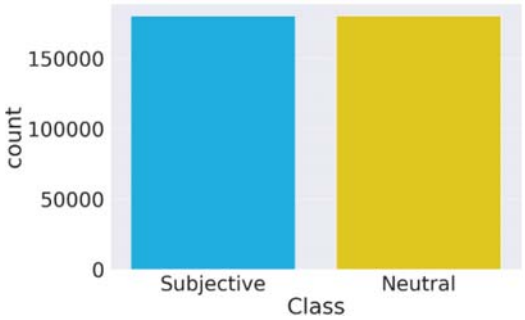

Fig. 4 Data class proportion

B. Results and Discussion

We built and trained our model on an experimental setup that utilized the Python environment, and Keras API with Tensorflow as backend utility. We employed Grid search technique to determine the best hyperparameters for our model, such as learning rate, dropout probability (which prevents overfitting) [22], etc. Since our model is built on BERT, we used the BERT tokenizer as a tokenization tool for the dataset. All computations were performed on a single RTX3090 GPU.

As seen from Table II, our model's performance is compared with existing methods. These existing methods are widely used for solving classification problems, and since this study is a classification task, we implement all models on the WNC dataset described in Subsection *A* as benchmark for result comparison. For simplicity, we present the results based on two major evaluation metrics: Accuracy and F1-Score.

1) Accuracy: is a popular evaluation indicator in classification tasks. To calculate accuracy, we divide the correctly classified samples by the total number of samples.

$$Accuracy = \frac{Correct\ Samples}{Total\ Samples} \tag{6}$$

2) F1-Score: is a function of the precision and recall of the test. Hence to determine the F1-Score, we first calculate the precision and recall. Where precision is the ratio of the correct predictions knowns as true positive (TP) to the sum of the correct predictions and the incorrect positive predictions knowns as false positive (FP). While, recall is the measure of the correct predictions from the sum of the incorrect negative predictions known as false negatives (FN) and the correct predictions.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

The F1-Score is then calculated as the harmonic mean of precision and recall.

$$F_1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{9}$$

Bi-LSTM using the Glove embeddings achieved the lowest performance with an accuracy of 81%. Followed closely by BERT sentence classifier which achieved an accuracy of 84%. Although, BERT + BiLSTM without attention produced impressive results, BERT + BiLSTM with attention (which is our proposed model) outperformed it by a margin of 0.03%, resulting to 0.89% accuracy. We recorded high precision and recall on some of the baselines, however, our proposed model consistently outperformed them across all metrics.

TABLE II
Result Comparison

Model	Accuracy	F1-score
BERT	0.84	0.87
BiLSTM + Glove	0.81	0.80
BERT + BiLSTM w/o Att	0.86	0.86
BERT + BiLSTM w/ Att (our model)	**0.89**	**0.90**

We also report performance results on the BiGRU version of our model, since BiGRU is also a variant of BiRNN which our model's BiLSTM layer is based off of. From that experiment (though not recorded in this paper), we find that BiGRU achieves a comparable performance to the proposed BiLSTM in a slightly shorter time. However, BiLSTM generalizes better to the dataset thereby producing better results. These results support our motivation for seeking an improvement to existing subjectivity detection techniques. Although we tested our model on the entire test set that was set aside before training, to visualize the importance and effectiveness of the attention mechanism, we utilized two sentences from the test set as input

to the model. Similarly, to test for robustness, we use two sentences from Wikipedia revisions of real-world samples as input to the model.

Integrating the attention visualization code implementation of [23] to our model, the resultant output in Fig. 5 is a visual representation of the attention applied to the words in each sentence and the corresponding label for that sentence.

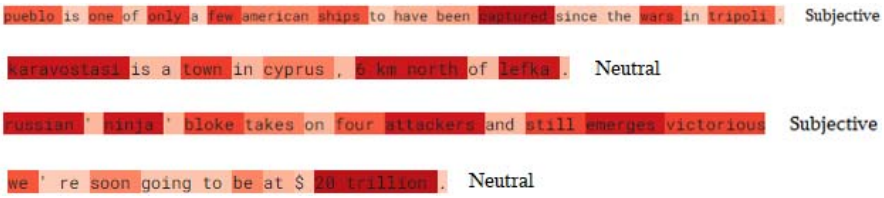

Fig. 5 Attention Visualization (The first two sentences were taken from the test set and the last two sentences were taken from Wikipedia revisions)

V. Conclusion

In this work, we examine what distinguishes subjective text from the neutral counterpart. We utilized an approach that combines BERT, BiLSTM, and attention mechanism (BERT + BiLSTM + Attention). Using BERT as the upstream enhances the performance of downstream model. Since this model is built to identify subjective bias in text, we used the WNC corpus to benchmark the model and compare it to previous approaches. And results show that the proposed model outperforms existing approaches by a clear margin, indicating an improved performance.

Future directions will be carried out towards refining our network to accommodate multilingual texts since this study only focused on English text representation and classification. We will also work towards document level subjective bias detection.

References

[1] Foundation, O. S. 2018. Indicators of news media trust. https://kf-site-production.s3.amazonaws.com/media_elements/files/000/000/216/original/KnightFoundation_Panel4_Trust_Indicators_FINAL.pdf.

[2] Gallup. 2018. Americans: Much misinformation, bias, inaccuracy in news. https://news.gallup.com/opinion/gallup/235796/\americansmisinformation-bias-\inaccuracy-news.aspx.

[3] Africanews: The voice of Africa. 2021. https://www.africanews.com/2021/01/04/jack-ma-s-disappears-from-african-tv-show-sparking-questions-over-wherabouts/

[4] Reid Pryzant, Richard Diehl Martinez, Nathan Dass, Sadao Kurohashi, Dan Jurafsky, and Diyi Yang. 2019. "Automatically Neutralizing Subjective Bias in Text," ArXiv Preprint arXiv:1911.09709.

[5] Wikipedia: Neutral point of view. (As updated on 13 March 2021). https://en.wikipedia.org/wiki/Wikipedia:Neutral_point_of_view

[6] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean (2013). "Efficient Estimation of Word Representations in Vector Space." arXiv:1301.3781 (cs.CL) 7 Sep 2013.

[7] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.

[8] Marta Recasens, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. 2013. "Linguistic Models for Analyzing and Detecting Biased Language. In Proceedings of the Association for Computer Linguistics," 1650–1659.

[9] Kartikey Pant, Tanvi Dadu, and Radhika Mamidi. 2020. "Towards detection of subjective bias using contextualized word embeddings," In Companion Proceedings of the Web Conference 2020, WWW 20, page 7576, New York,NY, USA. Association for Computing Machinery.

[10] N. Yu, and S. Kübler, 2011. "Filling the gap: Semi-supervised learning for opinion detection across domains," In Proceedings of the 15th Conference on Computational Natural Language Learning (CoNLL 2011) (pp. 200– 209).

[11] J. Wiebe and E. Riloff. 2005. "Creating subjective and objective sentence classifiers from unannotated texts," In Proceedings of the Conference on Computational Linguistics and Intelligent Text Processing (CICLing), volume 3406, pages 486–497. Springer.

[12] W. Li, D. Li, H. Yin, L. Zhang, Z. Zhu, P. Liu, "Lexicon-Enhanced Attention Network Based on Text Representation for Sentiment Classification," *Appl. Sci. 2019, 9*, 3717. https://doi.org/10.3390/app9183717.

[13] Desislava Aleksandrova, François Lareau, Pierre André Ménard. "Multilingual sentence-level bias detection in Wikipedia," 2019. Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2019) Varna, Bulgaria, 2019, pp. 42–51, doi: 10.26615/978-954-452-056-4_006.

[14] Aniruddha Ghosh and Tony Veale. "Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal," In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 482–491, 2017.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: PreTraining of Deep Bidirectional Transformers for Language Understanding." ArXiv Preprint ArXiv:1810.04805, 2018.

[16] R. Cai *et al*., "Sentiment Analysis About Investors and Consumers in Energy Market Based on BERT-BiLSTM," in IEEE Access, vol. 8, pp. 171408-171415, 2020, doi: 10.1109/ACCESS.2020.3024750.

[17] D. Liu, Z. Zhao and L. Gan, "Intention Detection Based on Bert-Bilstm in Taskoriented Dialogue System," 2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing, Chengdu, China, 2019, pp. 187-191, doi: 10.1109/ICCWAMTIP47768.2019.9067660.

[18] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," in IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673-2681, Nov. 1997, doi: 10.1109/78.650093.

[19] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. "Neural machine translation by jointly learning to align and translate," arXiv preprint arXiv:1409.0473, 2014.

[20] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[21] Adela Randall. 2017. CS 388: Natural Language Processing: LSTM Recurrent Neural Networks. https://slideplayer.com/slide/12965275/.

[22] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, 2014. "Dropout: a simple way to prevent neural networks from overfitting," The Journal of Machine Learning Research 15(1):1929–1958.

[23] Shreydesai: attention-viz. 2019. https://github.com/shreydesai/attention-viz