LEXICAL EVALUATION OF EXPLANATION PROVIDED BY ATTENTION MECHANISMS IN LSTM NETWORKS

SEBASTIAN PERALTA [PERALTAS@SEAS.UPENN.EDU], YILIN GENG [LINK10@SEAS.UPENN.EDU], JOHN BELLWOAR [BELLWOAR@SEAS.UPENN.EDU]

ABSTRACT. Recent attention interpretability measures fail to move beyond the internal mechanics of a model which can be undesirable when explaining the decision-making process in cross-disciplinary situations. In this paper we propose a lexica-based approach to measure a model's interpretability that makes use of the publicly available LIWC dictionary. As LIWC is annotated by humans and constructed to reflect human sentiment [3], this approach has the potential for increased transparency over more commonly used measures. We apply this approach on a vanilla LSTM and a diversity LSTM proposed by (Mohankumar et al., 2020) [2] and compare it to other commonly-used interpretability measures in the literature [6]. Our results show that using this lexica-based approach provides more information about a model's interpretability, is more transparent to the decision making, and aligns better with human understanding over commonly-used metrics that focus on the internal workings of a model for the vanilla and diversity LSTM.

1. INTRODUCTION

Apart from improving the performance of deep neural networks, attention mechanisms are often used to provide insights on how and why they make predictions. There is an ongoing debate on whether the learned weights of an attention layer can be considered sufficient explanation of the predictions. Previous studies show that attention weights are not as interpretable as we imagined them to be. For instance, manipulating the weights do not change the predictions much [1], attention weights cannot identify tokens that are relevant to the final decision [5], and attention layers in some tasks merely function as a gating unit [6]. Meanwhile, other work refutes the claims that attention does not provide explanation and show it is still possible that we can use attention mechanisms as explanations of network predictions. [7].

This paper further explores the interpretability of attention models by building upon the work of [2] and [4]. Our primary contribution is the introduction of a lexicon-based analysis method to evaluate attention-based models.By both visualizing the attention distribution over words, and comparing whether the given attention scores from the learned model correspond to "real world" perceptions of sentiment, we verify the explainability of attention models beyond the internal mechanics of the network. We first reproduce the the diversity LSTM from [2] and compare some of the interpretability metrics in [6] to a vanilla LSTM. We focus on the diversity LSTM as it scores favorably among the common interpretability measures. We then outline our lexica-based approach on the diversity LSTM and vanilla LSTM for binary classification tasks and discuss its usefulness as a measure of interpretability.

This paper proposes an approach to model human understanding of words sentiments as one-dimensional coordinates projected from word embedding space using pre-defined dictionary LIWC2015 [3]. This allows us to quantitatively evaluate interpretability by comparing the lexica scores generated from attention to the projection coordinates.

2. RELATED WORK

There has been substantial work on understanding attention in neural network models. While seemingly contradictory viewpoints have emerged about the interpretability of attention weights [1], one paper by Mohankumar claims [2] that interpretability can be improved by orthogonalizing the hidden states by either forcing them to be orthogonal to the mean of previous ones or adding a term that decrease the similarity (conicity). This second model, called a diversity LSTM, which is the focus of this paper, is trained using $L(\theta) = -p_{model}(y|P, Q, \theta) + \lambda\, conicity(H^P)$, where P and Q are the input sentences and H^P contains the hidden states of the LSTM. However their measurements of interpretability range from attention weight perturbance and evaluation to annotation of the decision making by random human subjects. The former has a strong focus on the internal mechanics of the model which may be undesirable in settings where the decision-making needs to be explained to a non-technical expert. The latter is unfeasible to perform in most circumstances. We reproduced two of these interpretability measures in our paper which are first outlined by Vashishth [6].

This paper proposes a lexical interpretability measure that is more transparent and provides more linguistic insights. Our work was inspired by the emotion analysis of text using a lexicon, where the lexicon is defined to be a set of informative words with associated weights or scores [4]. Compared to black-box deep learning models, lexicon-based models provide more interpretative predictions.

3. Approach

3.1. **Reproduction and Setup.** Working with the code made public by [2], we constructed a pipeline capable of training and running experiments and analysis on a vanilla LSTM and a diversity LSTM as defined and built in [2]. These experiments and analysis explore whether the diversity mechanism does indeed improve interpretability in the notions of plausibility and faithfulness.

3.2. **Feature Importance by Weight Manipulation.** The following experiments were run on a vanilla LSTM and a diversity LSTM on the IMDB and Yelp datasets.

3.2.1. *Random Attention Weights.* Here we compared the test accuracy of the diversity model with the test accuracy of the same model but with randomized attention weights. The weights were randomized by resampling from a normal distrubtion parameterized by the deterministic mean and variance. We do the same with the vanilla LSTM and compare.

3.2.2. *Random Permutation of Attention Weights.* Here we created 10 diversity models that are identical except they have randomly permuted attention weights. We then compared the average test accuracy of these random permutations with the test accuracy of the original diversity model. This experiment differs from the previous experiment as it is meant to measure the plausibility of the model defined as the ability for the network to provide a plausible reconstruction of the decision-making [2]. We repeat this process for the vanilla LSTM.

3.3. **Lexical analysis.** This was inspired by the idea to measure similarity/distance in the embedding space usually used for medical images. To embed the words we used fastText, which captures relationships between words and embeds similar words with similar representations. The similarities between vectors of similar words allows for a similarity score between words to be calculated using cosine similarity and their L2 distance. Individual words were embedded using a pre-trained, publicly available model of word vectors trained on Wikipedia. For the tokens with high attention weights, their distribution in this 300-dimensional embedding space might provide some explanations for the model.

The task of our models here were binary classifications. To have consistent evaluation across different datasets, we chose three datasets with similar labels, IMDB, Yelp, and Stanford Sentiment Treebank (SST). Although they were about different topics (movies, restaurants and sentiments), the labels were still binary. In other words, models trained on these datasets would predict how positive or negative a given text data is.

One of the difficulty to evaluating tokens with high attention weights in the embedding space was that attention does not provide 'direction', it reflects extremely positive and negative tokens equally without being able to tell you which is which.

As a result, we decided to obtain two lexica for each dataset where each lexicon contains all unique tokens in the dataset with a score indicating the average attention weights associated with the tokens. The attention scores of the first lexicon was only evaluated on positive reviews and the scores of the second lexicon on negative reviews. Each score is computed by averaging over the attention outputs of samples containing a token.

We inspected them separately and also subtracted the negative scores from the positive scores. The subtraction trick was expected to get rid of the neutral background tokens, such as 'movie' in the IMDB dataset which would have high attention weights in both positive data and negative data. The resulting lexica contain the tokens with a score indicating how much the token contributes to a positive or negative prediction with positive score indicating positive impacts.

If attention mechanism provides explanation, then the score will correlate with how positive the words/tokens in real-world language use. In that case, tokens with high score would be the words people use when expressing positive attitudes and tokens with very negative score would be those for negative feelings. If we pick a few words with highest and lowest scores, they should be separated clearly in the embedding space.

We first tried clustering methods including k-means++ with and without PCA on the lexica in the embedding space, but it did not result in clear clusters. We therefore decided to project the word vectors to a lower dimensional space manually with a dictionary-based approach. This was done by taking two word lists in the LIWC dictionary [3] corresponding to categories of 'positive emotion' and 'negative emotion' and computing the mean vector for each word list. We then computed the cosine-similarity and the L2 distance between a token and the two mean vectors, the difference of the similarity gives us a coordinate in this 1-dimensional projection. A positive word was expected to be

close to the mean vector of positive words defined in LIWC and far from the negative mean vector, hence the coordinate would be a more positive value. In the same vain, a negative word will give a more negative negative value.

This projection works as an estimate of how positive a word is. As a result, by comparing the distributions of the attention score and dictionary score over lexica, we can observe how much explanation the attention mechanism is providing.

4. Experimental Results

Experiments were run on the IMDB Movie Reviews, Yelp, and Stanford Sentiment Analysis Treebank (SST) datasets and consisted of sentiment analysis binary classification into positive and negative groups. For each dataset two models were trained— a vanilla LSTM model with a standard attention mechanism and a diversity model LSTM with modified loss function that decreases the conicity of hidden states.

4.1. **Feature Importance by Weight Manipulation.**

The two experiments we performed were randomizing and randomly permuting the weights of the attention layer. If there is a significant difference in the performance of the model when the weights are randomized, this means that the values of the weights are vital to model performance. If there is a significant difference in the performance of the model when the weights are permuted, this means that the decisions made by this model are plausible because the weight values are particular to the hidden states.

	IMDB		Yelp	
Weight Alteration	Vanilla	Diversity	Vanilla	Diversity
No alteration	0.8752	0.8634	0.9486	0.9357
Randomly permute	0.8701	0.5782	0.8701	0.6866
Randomly set	0.8752	0.5066	0.8752	0.8108

FIGURE 1

Figure 1 reports the accuracy of the diversity and vanilla LSTM models on the IMDB and Yelp datasets, as well as the accuracies of the two randomized experiments for each pair of model and dataset. We note that for the unaltered models, the vanilla LSTM model and the diversity LSTM model perform similarly for both data sets. However, randomly permuting the weights appears to penalize the diversity LSTM more than the vanilla LSTM for both datasets. Randomly setting the weights does likewise for the IMDB dataset but only slightly for the Yelp dataset. For the most part these results indicate higher plausibility and importance of these attention weights in the diversity model however the decision-making process remains elusive outside of the internal workings of the model.

4.2. **Lexica Analysis.**

For each model and classification task, the top 100 words with highest attention scores were chosen for positively and negatively classified reviews. The difference in similarity between each word in these sets and the mean positive/negative vectors of LIWC were then calculated in the manner described previously. The resulting distributions, individual means, and combined standard deviations for the Yelp dataset can be seen in Figure 2. Similar trends for other datasets can be seen in Figure 3 in the appendix.

For both the L2 distance and cosine similarity measures of similarity, the diversity LSTM performed as hypothesised and resulted in a more positive mean for the lexicon of the positive sentences, a more negative mean for the lexicon of the negative sentences, and a larger standard deviation for the combined lexicon when compared to the baseline vanilla LSTM. These trends held true for all datasets and lexica analyzed for diversity models compared to vanilla models. These trends reflect the expectation that the diversity model will place attention weights onto more positively and negatively charged words given its classification task. The mean of the positive list distribution becoming more positive for the diversity model reflects the fact that the weights were placed on words that were closer in space and had a greater cosine similarity to our mean positive word vector obtained from the LIWC dictionary. The same is true for the negative lists. Figure 2 also shows that the distributions became more spread-out with reduced peaks around zero. This is because there are less neutral words in the word lists. These neutral words include articles and other words one would not place high attentions weights in a positive/negative classification task. This was confirmed with manually viewing the highest weighted words in the word lists of vanilla and diversity models. This observation is confirmed by the fact that the standard deviation of the combined positive and negative lists increased for the diversity LSTMs. An increased standard deviation means that the overall list contains more positively and negatively correlated word, spreading out the overall distribution.

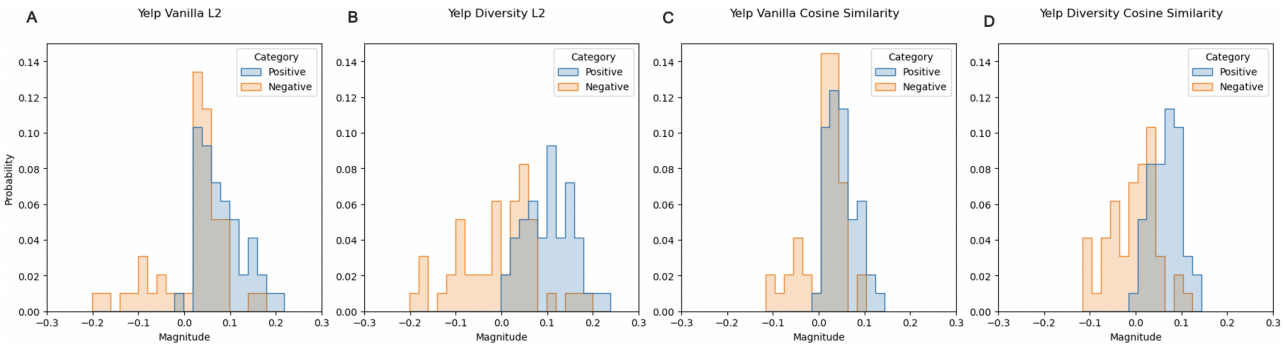

FIGURE 2. Histogram plots of the resulting distributions for L2 distance or cosine similarity calculations for the top 100 attention words for positively and negatively classified sentences without subtraction for the Yelp dataset. A) Vanilla LSTM L2 Distance Results. Positive Mean: 0.0799, Negative Mean: 0.0180, Standard Deviation: 0.0706. B) Diversity LSTM L2 Distance Results. Positive Mean: 0.1034, Negative Mean: -0.0085, Standard Deviation: 0.0911. C) Vanilla LSTM Cosine Similarity Results. Positive Mean: 0.0491, Negative Mean: 0.0105, Standard Deviation: 0.0437. D) Diversity LSTM Cosine Similarity Results. Positive Mean: 0.0646, Negative Mean: -0.0061, Standard Deviation: 0.0563

One observation is that lexica tend to be more positive than negative. The mean of the negative distributions tended to be closer to zero than the positive distributions for all models. Inspection of the attention words reveals that words generally deemed "positive" still occur frequently in the negative lists. One explanation for this is that attentions should not simply be viewed as keys or scores that themselves determine the classification of a sentence, but need to be understood in the context of a sentence. For example "not good" becomes negative despite "good" being considered a positive word and therefore receives a high attention score in both positive and negative lists. Also the tendency of the mean of the negative list to be closer to zero than the positive list could reflect a tendency of English to use these modifiers on positive words to turn them negative more often than the inverse. A subtraction method of reducing the overlap between the positive and negative sets for each model was conducted and found to generally reduce the mean of the negative word lists, but overall it had minimal impact on the distributions, their statistics, or the trends between vanilla and diversity models.

We could get a more intuitive understanding of the explanation provided by attention weights by manually inspecting the words (tokens) with highest weights or scores after subtraction. We noticed that the tokens in the lexica were a mixture of upper and lower case letters and punctuation was also mixed with words. We suspected a flaw in the tokenization of the implementation of the authors. This could be problematic and significantly compromise the resulting lexica, since if the same word in different forms was recognized as multiple unique tokens, the attention weights on them were no longer a faithful reflection of how much the word affected the prediction.

We fixed the problem by using a Spacy tokenizer and converting all tokens to lower case in the preprocessing. The resulting lexica aligned much better with human intuition. For example, the most positive words considered by the diversity LSTM model in the Yelp dataset are 'great, good, best, love, amazing, delicious' and punctuation '!', while the most negative ones are 'not, horrible, worst, terrible, rude, bad'. Since the time was limited and each experiment takes hours to finish with GPU resources, we only reran the experiments over again for the Yelp dataset.

By human inspection, the words that were considered positive and negative by the attention weights were the words commonly seen in positive and negative reviews for restaurants as shown in the Appendix. They include 1) adjectives that directly modify positive and negative objects, for example "good, great, nice, amazing, bad, worst, horrible"; 2) words describing specific aspects of a restaurant, 'delicious, fresh, tasty' for the food, 'friendly, professional, fast, rude, slow' for the service, 'clean, beautiful, dirty' for the environment. We can also see words that are not explicitly positive or negative but have clear association, for example, 'recommend' in positive comments, 'minutes, money, waste' in negative comments often complaining about slow service and high price.

The word lists allow us to observe the impact of increasing conicity of hidden states on the explanation provided by attention weights. By comparing the results of the diversity LSTM and vanilla LSTM with the subtraction trick on the fixed Yelp dataset, we would see that the word lists generated by the diversity LSTM pay less attention to background words and punctuation. The word lists also demonstrated the effect of our subtraction trick. If we look at the results with

and without subtraction trick for the fixed Yelp dataset, we could observe that the subtraction trick removed background words like 'but, food, back'.

5. Discussion

The attention weight manipulation experiments as performed in previous work on interpretability [2], [6] provide a loose measurement of of a models interpretability and ability to capture its decision-making process. However, the amount of information we can extract about the interpretability is limited and the explanation of the interpretability does not move beyond the internal mechanics of the model.

By modeling human intuition of word usage as the distribution of a pre-defined dictionary (LIWC [3]) in the embedding space, we came up with quantitative way to measure how much explanation is provided by the lexica generated using attention mechanism. The FastText embedding of this generated lexica and the pre-defined dictionary into a higher dimensional space allows use to directly compare the two. Both the L2 distance and cosine similarity proved meaningful measures of similarity. Overall we found that for the simple positive/negative binary classification task, the attention lexica corresponds well to human intuition and manual inspection of word lists, which indicates that the attention mechanism in the task provides explanation at least at token level. We also found the implementation of the diversity LSTM provided better correlation to human intuition both in a qualitative analysis of the word list and our quantitative lexica analysismeaning increasing conicity of hidden states had positive impact on interpretability of attention. The benefit of using lexica is that lexica themselves are more interpretable than the weights since they could also be inspected from linguistics angle directly, unlike weights in complex models.

Future work could explore the use of these generated lexica directly for prediction. A potential example is an algorithm that sums up the score of the tokens in a sentence to determine whether the sentence is positive or negative in general. This could possibly even produce a better generalization error over current high-complex models. Other work could experiment with the diversity hyper-parameter in the loss function or with making use of the constructed lexica from the LIWC dictionary to modify the loss function so the attention score distribution is better correlated with the lexica score distribution.

REFERENCES

[1] Sarthak Jain and Byron C. Wallace. Attention is not explanation, 2019.
[2] Akash Kumar Mohankumar, Preksha Nema, Sharan Narasimhan, Mitesh M. Khapra, Balaji Vasan Srinivasan, and Balaraman Ravindran. Towards transparent and explainable attention models, 2020.
[3] Booth R.J. Boyd R.L. Francis M.E. Pennebaker, J.W. *Linguistic Inquiry and Word Count: LIWC2015*. University of Texas at Austin, 2015.
[4] João Sedoc, Sven Buechel, Yehonathan Nachmany, Anneke Buffone, and Lyle Ungar. Learning word ratings for empathy and distress from document-level user responses, 2020.
[5] Sofia Serrano and Noah A. Smith. Is attention interpretable?, 2019.
[6] Shikhar Vashishth, Shyam Upadhyay, Gaurav Singh Tomar, and Manaal Faruqui. Attention interpretability across nlp tasks, 2019.
[7] Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation, 2019.

6. APPENDIX

IMDB					
	Measure	Model	Positive Mean	Negative Mean	Standard Deviation
Not Substracted	L2	Vanilla	0.0704	0.0060	0.0833
	L2	Diversity	0.0906	-0.0151	0.1020
	Cosine Similarity	Vanilla	0.0421	0.0027	0.0507
	Cosine Similarity	Diversity	0.0564	-0.0093	0.0626
Substracted	L2	Vanilla	0.0817	-0.0255	0.0849
	L2	Diversity	0.0899	-0.0429	0.0943
	Cosine Similarity	Vanilla	0.0513	-0.0163	0.0529
	Cosine Similarity	Diversity	0.0564	-0.0270	0.0589
Yelp					
	Measure	Model	Positive Mean	Negative Mean	Standard Deviation
Not Substracted	L2	Vanilla	0.0799	0.0180	0.0706
	L2	Diversity	0.1034	-0.0085	0.0911
	Cosine Similarity	Vanilla	0.0491	0.0105	0.0437
	Cosine Similarity	Diversity	0.0646	-0.0061	0.0563
Substracted	L2	Vanilla	0.0922	-0.0146	0.0825
	L2	Diversity	0.1079	-0.0326	0.0928
	Cosine Similarity	Vanilla	0.0584	-0.0097	0.0524
	Cosine Similarity	Diversity	0.0686	-0.0216	0.0590
SST					
	Measure	Model	Positive Mean	Negative Mean	Standard Deviation
Not Substracted	L2	Vanilla	0.0553	0.0190	0.0509
	L2	Diversity	0.0860	-0.0028	0.0745
	Cosine Similarity	Vanilla	0.0334	0.0099	0.0322
	Cosine Similarity	Diversity	0.0528	-0.0033	0.0466
Substracted	L2	Vanilla	0.0663	-0.0159	0.0744
	L2	Diversity	0.0913	-0.0243	0.0854
	Cosine Similarity	Vanilla	0.0414	-0.0112	0.0471
	Cosine Similarity	Diversity	0.0576	-0.0165	0.0540

FIGURE 3. Statistics of the distributions for L2 distance or cosine similarity calculations for the top 100 attention words for positively and negatively classified sentences without subtraction for the IMDB, Yelp, and SST datasets

	Yelp: Top 10 Words							
	Original				Cleaned			
	No subtraction		Subtraction		No subtraction		Subtraction	
	Vanilla	Diversity	Vanilla	Diversity	Vanilla	Diversity	Vanilla	Diversity
Positive Words	and	I	great	great	.	great	!	great
	I	and	Great	Great	!	good	great	good
	the	great	best	and	great	!	.	!
	great	good	love	good	and	best	and	best
	good	Great	and	best	,	love	good	love
	The	the	good	love	good	amazing	love	amazing
	this	The	always	this	food	the	amazing	delicious
	Great	This	is	I	i	friendly	,	friendly
	best	best	favorite	friendly	this	delicious	place	recommend
	is	love	recommend	nice	the	nice	delicious	nice
Negative Words	the	not	worst	not	.	not	?	not
	I	I	but	but	!	n't	worst	n't
	and	but	rude	worst	and	horrible	horrible	horrible
	The	The	back.	Not	i	worst	rude	worst
	but	the	not	no	,	good	terrible	terrible
	this	and	horrible	don't	but	no	ok	rude
	good	This	bad	never	back	the	bad	bad
	food	to	Worst	bad	the	terrible	disappointed	no
	to	Not	customer	rude	food	rude	bland	disappointed
	This	no	Horrible	didn't	this	bad	minutes	ok

FIGURE 4. Top 10 words with highest attention weights for different models on the Yelp dataset

Yelp Diversity LSTM		
	Positive Words	Negative Words
1	great	not
2	good	n't
3	!	horrible
4	best	worst
5	love	terrible
6	amazing	rude
7	delicious	bad
8	friendly	no
9	recommend	disappointed
10	nice	ok
11	favorite	minutes
12	awesome	bland
13	excellent	never
14	worth	mediocre
15	definitely	overpriced
16	the	poor
17	fresh	nothing
18	loved	awful
19	fantastic	slow
20	perfect	disappointing
21	tasty	wo
22	fun	dirty
23	clean	dry
24	wonderful	average
25	happy	told
26	yummy	money
27	:)	?
28	enjoyed	worse
29	helpful	sucks
30	vegas	disgusting
31	reasonable	waste
32	will	too
33	spot	away
34	professional	cold
35	fast	okay
36	cool	meh
37	outstanding	over
38	beautiful	better
39	come	else
40	yum	gross

FIGURE 5. Top 40 word with highest attention weights for the Diversity LSTM on the Yelp dataset

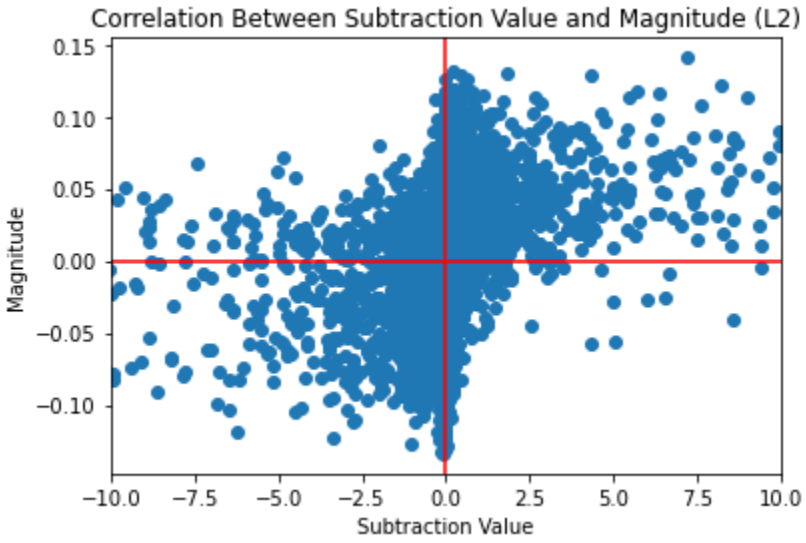

FIGURE 6. Graph of correlation between the subtracted attention values and magnitude of the L2 similarity calculation on the Yelp dataset