BEYOND WORD IMPORTANCE: CONTEXTUAL DECOMPOSITION TO EXTRACT INTERACTIONS FROM LSTMS

-W. James Murdoch, Peter J. Liu, Bin Yu

January 24, 2020

Presenter: Rishab Bamrara

https://qdata.github.io/deep2Read/

Motivation

LSTMs are successful because of their ability to learn complex and non-linear relationships. However, we are unable to describe the learned relationships of LSTMs which has led to LSTMs being characterized as black boxes.

Background

LSTMs are a core component of neural NLP systems. Given a sequence of word embeddings $x_1, \ldots\ldots, x_T \in R^{d_1}$, a cell and state vector $c_t, h_t \in R^{d_2}$ are computed for each element by iteratively applying the below equations, with initialization $h_o = c_o = 0$.

$$o_t = \sigma(W_o x_t + V_o h_{t-1} + b_o) \tag{1}$$
$$f_t = \sigma(W_f x_t + V_f h_{t-1} + b_f) \tag{2}$$
$$i_t = \sigma(W_i x_t + V_i h_{t-1} + b_i) \tag{3}$$
$$g_t = \tanh(W_g x_t + V_g h_{t-1} + b_g) \tag{4}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{5}$$
$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

Where $W_o, W_i, W_f, W_g \in R^{d1xd2}$, $V_o, V_f, V_i, V_g \in R^{d2xd2}$, $b_o, b_i, b_f, b_g \in R^{d2}$ and $\odot$ represents element-wise multiplication. o_t, f_t and i_t are often referred to as output, forget and input gates and their value lies in between 0 and 1.

Background (Contd.)

After processing the full sequence, the final state h_T is treated as a vector of learned features, and used as an input to SoftMax logistic regression, to return a probability distribution p over C classes, with:

$$p_j = \text{SoftMax}(Wh_T)_j = \frac{\exp(W_j h_T)}{\sum_{k=1}^{C} \exp(W_k h_t)} \tag{7}$$

Related Work

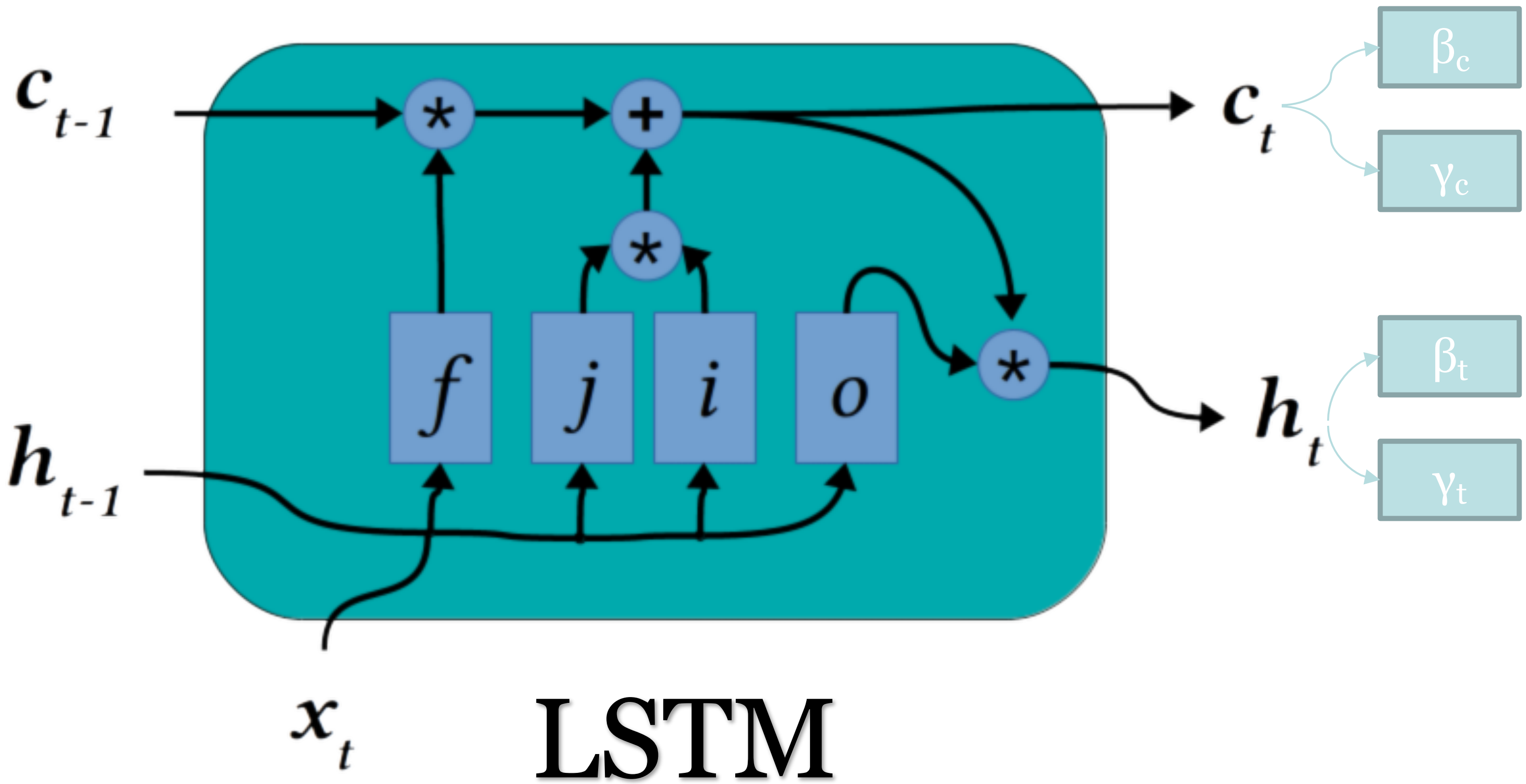
Mostly the previous work on interpreting LSTMs has focused on approaches for computing word-level importance scores, with varying evaluation protocols.

- Murdoch & Szlam (2017): Cell Decomposition.
- Li et al. (2016): Leave One Out.
- Sundararajan et al. (2017): Integrated Gradients.
- Karpathy et al. (2015), Strobelt et al. (2016): Analysing gate activations.
- Bach et al. (2015), Shrikumar et al. (2017): Applied Decomposition-based approaches to CNNs.
- Bahdanau et al. (2014): Attention based models.

Claim / Target Task

Without changing the underlying model of LSTM and decomposing it's output, CD (Contextual Decomposition) captures the contributions of combinations of words or variables to the final prediction of LSTM.

An Intuitive Figure Showing WHY Claim



https://blog.exxactcorp.com/5-types-lstm-recurrent-neural-network/

Proposed Solution

Given an arbitrary phrase x_q, ..., x_r, where 1<= q <= r <= T, decompose each output and cell state c_t, h_t in equations 5 and 6 (above) into a sum of two contributions:

$$h_t = \beta_t + \gamma_t$$
$$c_t = \beta_c + \gamma_c$$

- β_t corresponds to contributions made solely by the given phrase to h_t, and
- γ_t corresponds to contributions involving, at least in part, elements outside of the phrase.
- β_c & γ_c are analogous to c_t.

Using the above decomposition, the final output state Wh_T is given as:

$$p = \mathrm{SoftMax}(W\beta_t + W\gamma_t)$$

Here, $W\beta_t$ provides a Quantative score for the phrase's contribution to the LSTM's prediction.

Implementation

Authors assume that they have a way of linearizing tanh and sigmoid gates and updates in equations 2, 3, 4.

$$i_t = \sigma(W_i x_t + V_i h_{t-1} + b_i) \tag{11}$$
$$= L_\sigma(W_i x_t) + L_\sigma(V_i h_{t-1}) + L_\sigma(b_i) \tag{12}$$

Once, we can do this then we can also linearize the element-wise inner product and hence find linearization for h_t and c_t.

$$f_t \odot c_{t-1} = (L_\sigma(W_f x_t) + L_\sigma(V_f \beta_{t-1}) + L_\sigma(V_f \gamma_{t-1}) + L_\sigma(b_f)) \odot (\beta^c_{t-1} + \gamma^c_{t-1}) \tag{13}$$
$$= ([L_\sigma(W_f x_t) + L_\sigma(V_f \beta_{t-1}) + L_\sigma(b_f)] \odot \beta^c_{t-1}) \tag{14}$$
$$+ (L_\sigma(V_f \gamma_{t-1}) \odot \beta^c_{t-1} + f_t \odot \gamma^c_{t-1})$$
$$= \beta^f_t + \gamma^f_t \tag{15}$$

Implementation (Contd.)

$$
\begin{aligned}
i_t \odot g_t = & [L_\sigma(W_i x_t) + L_\sigma(V_i \beta_{t-1}) + L_\sigma(V_i \gamma_{t-1}) + L_\sigma(b_i)] && (16)\\
& \odot [L_{\tanh}(W_g x_t) + L_{\tanh}(V_g \beta_{t-1}) + L_{\tanh}(V_g \gamma_{t-1}) + L_{\tanh}(b_g)] \\
= & [L_\sigma(W_i x_t) \odot [L_{\tanh}(W_g x_t) + L_{\tanh}(V_g \beta_{t-1}) + L_{\tanh}(b_g)] && (17)\\
& + L_\sigma(V_i \beta_{t-1}) \odot [L_{\tanh}(W_g x_t) + L_{\tanh}(V_g \beta_{t-1}) + L_{\tanh}(b_g)] \\
& + L_\sigma(b_i) \odot [L_{\tanh}(W_g x_t) + L_{\tanh}(V_g \beta_{t-1})]] \\
& + [L_\sigma(V_i \gamma_{t-1}) \odot g_t + i_t \odot L_{\tanh}(V_g \gamma_{t-1}) - L_\sigma(V_i \gamma_{t-1}) \odot L_{\tanh}(V_g \gamma_{t-1}) \\
& + L_\sigma(b_i) \odot L_{\tanh}(b_g)] \\
= & \beta_t^u + \gamma_t^u && (18)
\end{aligned}
$$

Now the decomposition of c_t can be found by summing the two contributions:

$$\beta_t^c = \beta_t^f + \beta_t^u \qquad (19)$$

$$\gamma_t^c = \gamma_t^f + \gamma_t^u \qquad (20)$$

Implementation (Contd.)

Once decomposition of c_t is computed, resulting transformation of h_t is given by:

$$
\begin{aligned}
h_t &= o_t \odot \tanh(c_t) && (21)\\
&= o_t \odot [L_{\tanh}(\beta_t^c) + L_{\tanh}(\gamma_t^c)] && (22)\\
&= o_t \odot L_{\tanh}(\beta_t^c) + o_t \odot L_{\tanh}(\gamma_t^c) && (23)\\
&= \beta_t + \gamma_t && (24)
\end{aligned}
$$

Linearization of tanh gate is also provided in the paper:

$$L_{\tanh}(y_k) = \frac{1}{M_N} \sum_{i=1}^{M_N} [\tanh(\sum_{j=1}^{\pi_i^{-1}(k)} y_{\pi_i(j)}) - \tanh(\sum_{j=1}^{\pi_i^{-1}(k)-1} y_{\pi_i(j)})] \tag{27}$$

Where, π_1, ..., π_{M_N} denote the set of all permutations of 1, ..., N variables inside the tanh gate excluding the bias term.

Data Summary

- **Stanford Sentiment Treebank (SST):** Standard NLP benchmark which consists of movie reviews ranging from 2 to 52 words long. In addition to labels of reviews, it also has labels for each phrase in the review.
- **Yelp Polarity:** This was obtained from the Yelp Dataset Challenge. It has train and test sets of sizes 560,000 ad 38,000 respectively. Average length of review is 160.1 words. It contains only review labels.

Experimental Results and Analysis

Model	SST (Accuracy)	Yelp (Error)
LSTM	87.2%	4.6%
Logistic Regression	83.2%	5.7%

Above results indicate that both LSTM and Logistic Regression perform well on SST as well as Yelp dataset.

Attribution Method	SST (Unigram scores)	Yelp (Error)
CD	0.76	0.52
Integrated Gradients	0.72	0.34 – 0.56
Other Methods	<=0.51	0.34 – 0.56

For SST, both CD and Integrated Gradients performs better out of all the other methods. On Yelp, although the gap is not very big, but CD is still very competitive and is closer to the best result. Overall, CD gives strong results.

Experimental Results and Analysis (Contd.)

Attribution Method	Kolmogorov-Smirnov one-sided test statistic
CD	0.74
Cell Decomposition	0
Integrated Gradients	0.33
Leave One out	0.58
Gradient	0.61

Kolmogorov-Smirnov one-sided test statistic is a common test for the difference of distributions with values ranging from 0 to 1. Larger the value means the method is able to identify strong difference between positive and negative distributions. As can be seen, CD outperforms all the other methods.

Experimental Results and Analysis (Contd.)

Attribution Method	**Heat Map**								
Gradient	used	to	be	my	favorite	not	worth	the	time
Leave One Out (Li et al., 2016)	used	to	be	my	favorite	not	worth	the	time
Cell decomposition (Murdoch & Szlam, 2017)	used	to	be	my	favorite	not	worth	the	time
Integrated gradients (Sundararajan et al., 2017)	used	to	be	my	favorite	not	worth	the	time
Contextual decomposition	used	to	be	my	favorite	not	worth	the	time

Legend: Very Negative | Negative | Neutral | Positive | Very Positive

Table 1: Heat maps for portion of yelp review with different attribution techniques. Only CD captures that ”favorite” is positive.

Experimental Results and Analysis (Contd.)

Attribution Method	Heat Map
Gradient	It's easy to love Robin Tunney – she's pretty and she can act – but it gets harder and harder to understand her choices.
Leave one out (Li et al., 2016)	It's easy to love Robin Tunney – she's pretty and she can act – but it gets harder and harder to understand her choices.
Cell decomposition (Murdoch & Szlam, 2017)	It's easy to love Robin Tunney – she's pretty and she can act – but it gets harder and harder to understand her choices.
Integrated gradients (Sundararajan et al., 2017)	It's easy to love Robin Tunney – she's pretty and she can act – but it gets harder and harder to understand her choices.
Contextual decomposition	It's easy to love Robin Tunney – she's pretty and she can act – but it gets harder and harder to understand her choices.

Legend Very Negative Negative Neutral Positive Very Positive

Table 2: Heat maps for portion of review from SST with different attribution techniques. Only CD captures that the first phrase is positive.

Experimental Results and Analysis (Contd.)

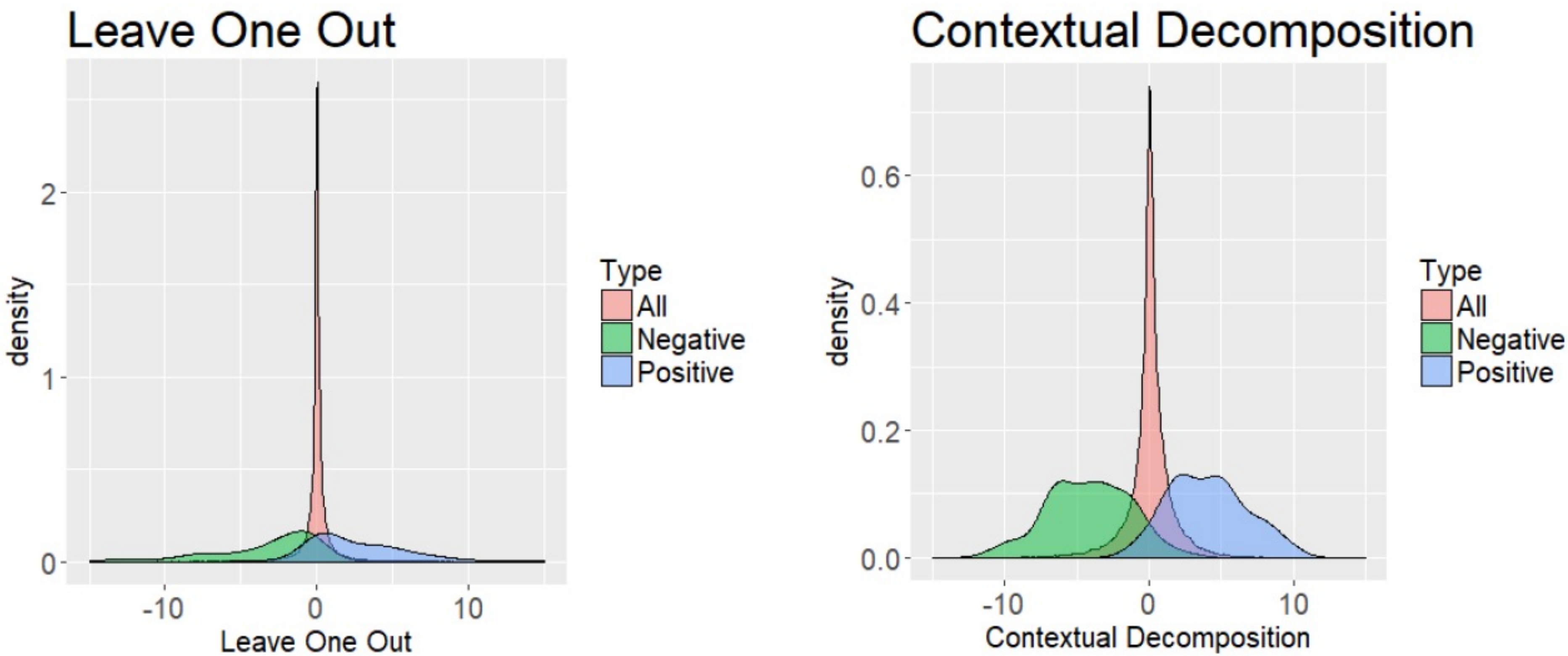


Figure 1: Distribution of scores for positive and negative negation coefficients relative to all interaction coefficients. Only leave one out and CD are capable of producing these interaction scores.

Experimental Results and Analysis (Contd.)

not entertaining	**not bad**	**very funny**	**entertaining**	**bad**
not funny	never dull	well-put-together piece	intelligent	dull
not engaging	n't drag	entertaining romp	engaging	drag
never satisfactory	never fails	very good	satisfying	awful
not well	without sham	surprisingly sweet	admirable	tired
not fit	without missing	very well-written	funny	dreary

Table 3: Nearest neighbours for selected unigrams and interactions using CD embeddings

Conclusion and Future Work

Proposed contextual decomposition (CD) algorithm is able to interpret predictions made by LSTMs without modifying the underlying model. In both NLP and general applications of LSTMs, CD produces importance scores for words, phrases and word interaction. CD also performs well in comparison with the other methods. Also, CD is capable of identifying phrases of varying sentiment and extracting meaningful word interactions.

References

- Sebastian Bach, Alexander Binder, Gr´egoire Montavon, Frederick Klauschen, Klaus Robert M¨uller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one, 10(7):e0130140, 2015.

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

- W James Murdoch and Arthur Szlam. Automatic rule extraction from long short term memory networks. ICLR, 2017.

- Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure. CoRR, abs/1612.08220, 2016. URL http://arxiv.org/abs/1612.08220.

- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. CoRR, abs/1703.01365, 2017. URL http://arxiv.org/abs/1703.01365.

- Andrej Karpathy, Justin Johnson, and Li Fei-Fei. Visualizing and understanding recurrent networks. arXiv preprint arXiv:1506.02078, 2015.