Attention Uncovers Task-Relevant Semantics in Emotional Narrative Understanding

Thanh-Son Nguyen[a], Zhengxuan Wu[b], Desmond C. Ong[a,c]

[a]*Agency for Science, Technology and Research (A*STAR), Singapore*
[b]*Symbolic Systems Program, Stanford University*
[c]*Department of Information Systems and Analytics, National University of Singapore*

Abstract

Attention mechanisms in deep neural network models have helped them to achieve exceptional performance at complex natural language processing tasks. Previous attempts to investigate what these models have been "paying attention to" suggest that these attention representations capture *syntactic* information, but there is less evidence for *semantics*. In this paper, we investigate the capability of an attention mechanism to "attend to" *semantically meaningful* words. Using a dataset of naturalistic emotional narratives, we first build a *Window-Based Attention* (WBA) consisting of a hierarchical, two-level long short-term memory (LSTM) with softmax attention. Our model outperforms state-of-the-art models at predicting emotional valence, and even surpassing average human performance. Next, we show in detailed analyses, including word deletion experiments and visualizations, that words that receive higher attention weights in our model also tend to have greater emotional semantic meaning. Experimental results using six different pre-trained word embeddings suggest that deep neural network models which achieve human-level performance may learn to place greater attention weights on words that humans find semantically meaningful to the task at hand.

Keywords: Explainable AI; Emotion Understanding; Neural Network Attention

1. Introduction

Deep Learning has achieved remarkable performance at complex natural language processing (NLP) tasks. One innovation in particular, neural network attention [4], has revolutionized the way deep language models solve tasks like machine translation [38], summarization [12], and emotion understanding [73, 72]. The most basic form of neural network attention involves learning a weighted sum of input tokens, for example, a weighted sum over the words (represented as vectors) of a sentence. More recent attention mechanisms attempt to learn more complicated sums: For example, self-attention, which has become the core of successful state-of-the-art models like the Transformer [63], and BERT [15], learns a weight for each pair of tokens. However, one main criticism of deep language models is their lack of interpretability or *explainability* [3]. In this paper, we specifically examine deep neural network attention within the context of an emotion understanding task, and investigate the capability of an attention mechanism to "pick out" *semantically meaningful* words.

In recent years, researchers have been interested in "peering into the black box" to try to understand what is it that attention mechanisms are paying attention to. Recent work has suggested that attention-based models like BERT [15] and GPT-2 [51] encode *syntactic information* [60], in that the learnt representations encode dependency relations between head words and their modifiers [13], encode distances in the sentence parse-trees [24], and can be used to identify syntactic categories like parts-of-speech [34, 64].

However, these efforts in studying attention and syntax stop short of showing evidence that attention mechanisms learn to pick out the *semantic information* necessary to solve specific linguistic tasks. Compared to semantics, syntactic information is more "basic" and is generalizable across a wider-variety of language tasks, and it is not surprising that analyses on attention mechanisms should first examine syntax. The next step in the language understanding pipeline [8] involves understanding *semantics*, which is task-specific. The semantics relevant to a translation task are rather different compared to that of an emotion-classification task. This highlights the difficulty of "probing" semantic understanding, and suggests that we need to start with a task that has relatively well-defined semantics. Additionally, it would be helpful if the chosen task semantics are also intuitive for human readers, to serve as a sanity check. For these reasons, we choose to analyze attention in an emotion understanding task—predicting the real-valued emotional valence of a speaker describing an emotional life event, i.e., a *narrative*.

As we are dealing with long narratives, a traditional Recurrent Neural Network (RNN) model is not a good option since one notable weakness of RNNs is their inability to handle long sequences, due to vanishing gradients [47] as they are propagated back through the recurrent connections. Therefore, inspired by [33], we chose to use a hierarchical (two-level) Long Short-Term Memory (LSTM) for our model architecture. In particular, one LSTM network is used to encode a short *window* of words (e.g., a sentence or a time-based window), and another LSTM network is used to encode the sequential information of the whole narrative. The advantage of this approach is that LSTM networks in the model do have to deal with too-long sequences, thus avoiding the issues including vanishing gradients, while still being able to take into account the context of the whole sequence.

Our contributions are as follows. First, we propose a *Window-Based Attention* (WBA) model which exploits a hierarchical RNN with a softmax attention mechanism to predict emotional valence. In particular, we use a two-level LSTM network to encode *local* and *global* sequential information of linguistic content. We apply an attention layer after the first LSTM to produce a local contextual encoding that is then passed to the second, global-level LSTM. We show that our proposed model outperforms state-of-the-art models on our chosen task, in fact, significantly surpassing human performance. Secondly, we present detailed analyses of attention weights showing that our attention layer tends to pick out words with high emotion semantics.

2. Related Work

2.1. Attention-based models for Emotion Understanding and Sentiment Analysis

The task we choose here, emotion understanding, is closely related to sentiment analysis [8], and is well-studied in NLP and Affective Computing [7]. From a social-scientific perspective, the main differences are that emotions are felt in response to an event (e.g., losing a loved one) and could be categorical (happy, sad) or lie along continuous dimensions (e.g., valence and arousal). On the other hand, sentiment refers to the attitude that someone has towards a target (e.g., do they feel positive about this product?), and is often just measured along a single negative-to-positive valence dimension. Here, our chosen task is predicting emotional valence, measured as a real-valued number conveying the "degree" of negativity or positivity felt by the speaker. Many previous papers have tackled predicting intensity of sentiment or emotions [56, 66]. For example, a recent paper [1] proposed a stacked ensemble method consisting of a multi-layer perceptron network which takes outputs of three deep learning models and a feature-based model to predict emotion and sentiment intensity.

Attention-based NLP models have proved to be very useful in emotion understanding [73]. Attention mechanisms such as self-attention [52] and cross-modal attention [31] have been applied for conversational emotion analysis, where the emotion depends on the utterance in context. In addition to text, attention-based models are commonly used for multimodal emotion understanding, using other modalities like speech [41, 78, 59] and vision [57, 67, 42]. Recent models have applied attention on top of previous established approaches, such as using Convolutional Neural Networks (CNNs) to extract features from audio spectrograms, and then learning an attention distribution over those features [48, 6].

Attention mechanisms have also been widely adopted for sentiment analysis [80], especially for *aspect-level* [68] and *multimodal* [25] sentiment analysis. In aspect-level sentiment analysis, the objective is to identify the sentiment for a specific *aspect* (e.g., a phone's *price* or its *battery life*). Previous models have used attention to learn the connection between an aspect and input content [68, 36, 5]. For example, [72] proposed a context-dependent attention mechanism, and [35] proposed a hierarchical attention, which both attempt to learn aspect-specific attention distributions. Attention-based models have also been used for multimodal sentiment analysis. For example, [30] and [25] proposed attention-based gating mechanisms to learn the relative importance of different modalities, such as text, audio and image, for predicting sentiment.

2.2. Previous model architectures

There are many recent NLP papers that use deep neural network models like LSTMs to achieve high performance for time-series emotion prediction in challenging datasets [79], including the dataset that we use [45, 73]. As we are dealing with long narratives, for our model architecture, we chose to use a hierarchical (two-level) LSTM. One notable weakness of RNNs is their inability to handle long sequences, due to vanishing gradients [47] as they are propagated back through the recurrent connections. There are two popular methods to address this limitation: The first uses 'gates' that allow longer-range dependencies (e.g., as in the LSTM network), while a second method uses a hierarchy such that the model learns dependencies at different time-scales.

To handle long sequences for document modeling, Lin et al. [33] introduced Hierarchical RNNs (HRNNs), which consists of two independent but nested RNNs, one at the word-level and one at the sentence-level, which improved performance on a machine translation task. HRNNs have also been used in text generation: Yu et al. [75] presented a HRNN that contains a sentence generator and a paragraph generator to generate paragraph captions for videos, and Krause et al. [29] used a HRNN to generate paragraph descriptions for images. HRNNs are also used for other modalities, such as speech recognition [9]. Most similar to our work is Yang et al. [74], who proposed a two-level HRNN with attention for document-level classification. One difference with our work is that our model predicts a real-valued label for each window instead of predicting a class for the whole document. Finally, Ma et al. [39] showed that a hierarchical LSTM with attention and commonsense knowledge performed well at aspect-based sentiment analysis, and presented some preliminary analyses on their attention weights. We go further and demonstrate that attention in our model captures emotion-relevant semantics.

2.3. Interpretability of Attention

One of the main contributions in this paper is the analysis of the attention layer. As we mentioned, attention mechanisms have shown to be effective at improving performance on NLP tasks such as machine translation [38, 4], reading comprehension [10] and semantic parsing [16]. Because of the exemplary performance of attention mechanisms, many researchers have attempted to analyze attention weights and what they actually learn, as a method to "explain" what the deep learning model is doing.

Earlier studies in attention have focused on investigating what attention does within the model. Cheng et al. [11] modified an LSTM model to perform language modelling using shallow reasoning with memory and local attention over hidden states, and showed, in a reading task, that attention tended to focus on recent "memory" states (i.e., the hidden representations of nearby tokens). Martins and Astudillo [40] introduce *sparsemax*, a new activation function similar to *softmax*, and show its capability of highlighting "key" words for making decision in a Natural Language Inference task. Some studies have also suggested that the attention weights may be human-interpretable. Pappas and Popescu-Belis [46] had human judges rate the importance of each sentence to the entire document (a product review), and found strong correlations between the ratings of human judges and the value calculated by machine attention in a document classification task. Most recently, Donkers et al. [17] provided sentences (from product reviews) which had received high attention weights, as explanations to support product recommendations, and provided preliminary survey evidence that users found such explanations as relevant and helpful.

It is also worth mentioning self-attention in Transformer-based models [63] such as BERT [15] and GPT-2 [51]. These self-attention distributions have been shown to encode or align with syntatic information such as part-of-speech, dependency relations and coreference [64, 60, 13]. Besides syntatic properties of Transformer-based models, one recent paper also showed that structured self-attention weights encode semantics across multiple sentiment analysis tasks [71]. Overall these papers, both for softmax attention (like the ones we study here) and self-attention, provide support for the idea that attention may have some interpretability.

On the other hand, there are also dissenting papers showing that attention weights are not (or only weakly) correlated with feature importance measures such as gradient-based methods [26, 55], and that there exist alternative "counterfactual" attention distributions that yield equivalent model predictions [26, 44, 50, 22]. In other words, the attention distribution learnt by a model is not unique, and one can generate alternative distributions that produce the same predictions—this would suggest that attention distributions are not learning anything "true" like the underlying semantics. Vashishth et al. [62] elaborated on these results, and suggest that attention weights are interpretable only in certain tasks (e.g., sequence-to-sequence tasks). Recently, Wiegreffe and Pinter [69] also presented issues with analyzing explanabilities of attention weights using "counterfactual" weights, and argued that such a claim was ambiguous in its definition of explanation and may only work in certain parts of the neural network. To increase the explainability of the attention weights, very recent frameworks have also been developed to explain attention weights such as attributing attention weights using a hierarchical tree structure [23], analyzing attention weights as a vector norm [28], and interpreting attention weights using a gradient update process [58].

While our paper does not claim to resolve this debate on what attention is or is not, we feel that our results do add to this debate. In this paper, we show that our attention layer is able to "pay more attention" to words that carry emotional semantic meaning. Furthermore, we show that the highly-attended words are essential for model's predictions by conducting *word deletion* experiments (Section 4.4).

Other methods of explanation. Although not studied in this paper, we feel it relevant to mention other methods of "explaining" the decision-making of deep learning models besides attention. These attribution methods are mostly gradient-based, and use different methods to calculate the importance of the model input to the model output, i.e., "∂-output / ∂-input". (We note that gradients are but one operationalization of "explanation", such as adopted by Jain and Wallace [26] to suggest that "attention is not explanation"). One important class of gradient-based methods, Layerwise Relevance Propagation (LRP), focuses on calculating the "relevance" between input features and the model outputs, and has been derived for various models like the LSTM [2] and Transformer [65]. LRP examines the contribution of each token in each layer of a model to each token in the subsequent layer. Thus, by going backwards from the model output back through the intermediate layers, one can *propagate* the *relevance* (i.e., the importance of each token) back through the layers. One important example to the context of this paper is Arras et al. [2], who used LRP to explain sentiment analysis by a bidirectional LSTM, by computing the relevance score of input words to the target label. They showed that relevance scores performed better at explaining model predictions than gradient-based sensitivity analysis, but they did not consider attention.

3. Window-based Attention Model

In this section we describe our window-based attention model which consists of two-level LSTMs with a *softmax* attention layer to predict emotional valence scores of a time-series narrative. We use $X = \{X^1, X^2, ..., X^n\}$ to denote a narrative consisting of n ordered *windows*. Each input window consists of a sequence of words $X^k = \{X_1^k, X_2^k, ..., X_{m_k}^k\}$, where m_k is the number of words in window X^k, $X_t^k \in \mathbb{R}^V$ is the

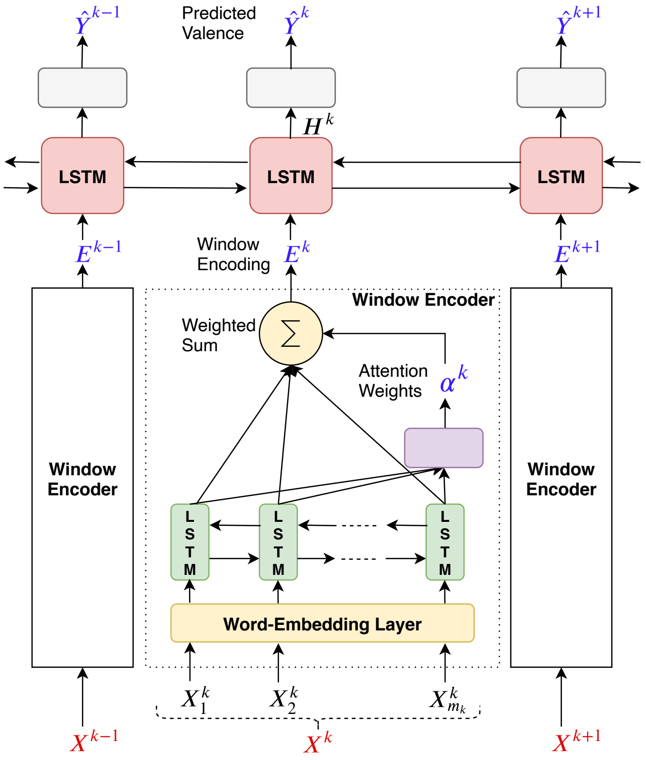


Figure 1: Our proposed Window-based Attention (WBA) model which includes a hierarchical (two-level) LSTM with attention mechanism to predict real-valued emotion valence in time-series. We implement both LSTMs using bidirectional LSTMs. In the text, for brevity, we present the equations for forward LSTMs.

one-hot vector of the t^{th} word in X^k, and V is the vocabulary size. The corresponding valence labels are denoted as $Y = \{Y^1, Y^2, ..., Y^n\}$, where $Y^k \in \mathbb{R}$ is the real-valued valence score for window X^k. Given an input narrative $X = \{X^k\}$, where $k = [1...n]$, the objective is to predict the valence score $\hat{Y}^k$ for each window $X^k \in X$.

Inspired by Lin et al. [33], we propose a WBA model which includes two levels of RNNs to encode the sequential information of each window (*local-level*) and the whole narrative (*global-level*) (Figure 1). Given a window X^k, WBA first encodes the window using its *window encoder*, which uses a LSTM network to compute sequential information of the words in the window, and a softmax attention layer to obtain the *window's encoding* ($\boldsymbol{E}^k$). The encoding is then passed to the WBA's global-level LSTM to aggregate local and global information before being given to the output layer to predict the valence score $\hat{Y}^k$.

We implemented bidirectional LSTM for both local-level and global-level LSTMs, as shown in Figure 1. When using a bidirectional LSTM, the hidden state for each time-step is the concatenation of the states of the forward LSTM and of the backward LSTM. For brevity, we present the equations for the forward LSTM only, to avoid doubling the number of equations and variables.

Window Encoder. Given a window $X^k = \{X^k_t\}$ where $t = [1...m_k]$, WBA computes the sequential information using the local-level LSTM in which the hidden state at step t is defined as:

$$\boldsymbol{h}^k_t = \psi\left(\boldsymbol{h}^k_{t-1}, \boldsymbol{f}_E(X^k_t|X^k); \boldsymbol{W}_\psi\right) \tag{1}$$

where $\boldsymbol{h}^k_t \in \mathbb{R}^d$ is the d-dimensional hidden state at step t; $\psi()$ and $\boldsymbol{W}_\psi$ are the local-level LSTM and its parameters, respectively; $\boldsymbol{f}_E$ is the function to transform one-hot vector to X^k_t to M-dimensional word embeddings, conditioned on the input window X^k (for contextual embeddings).

WBA computes the attention weights $\boldsymbol{\alpha}^k = \{\boldsymbol{\alpha}^k_t\}$ based on the attention mechanism introduced in [38]. The attention weight for the t^{th} word is computed by:

$$\boldsymbol{\alpha}^k_t = \frac{\exp(\boldsymbol{s}^k_t)}{\sum_{j=1}^{m_k} \exp(\boldsymbol{s}^k_j)} \tag{2}$$

where the score $\boldsymbol{s}^k_t$ is computed as follows:

$$\boldsymbol{s}^k_t = \boldsymbol{v}_a^\top \mathit{tanh}(\boldsymbol{W}_a \boldsymbol{h}^k_t) \tag{3}$$

where $\boldsymbol{W}_a$ and $\boldsymbol{v}_a$ are trainable parameters. The attention weights $\boldsymbol{\alpha}^k_t$ are used to calculate the window encoding $\boldsymbol{E}^k$ by weighting the hidden states:

$$\boldsymbol{E}^k = \sum_{t=1}^{m_k} \boldsymbol{\alpha}^k_t \boldsymbol{h}^k_t \tag{4}$$

where m_k is the number of words in window X^k. The window encoding $\boldsymbol{E}^k$ is passed to the global-level LSTM to predict the output valence score.

Valence Prediction. The global-level LSTM takes the window encoding $\boldsymbol{E}^k$ and computes its hidden state $\boldsymbol{H}^k$:

$$\boldsymbol{H}^k = \Psi\left(\boldsymbol{H}^{k-1}, \boldsymbol{E}^k; \boldsymbol{W}_\Psi\right) \tag{5}$$

where $\boldsymbol{H}^k \in \mathbb{R}^D$ is the D-dimensional hidden state of the global-level LSTM at step k; Ψ and $\boldsymbol{W}_\Psi$ are the global-level LSTM and its parameters. We base the implementation of our LSTMs $\Psi()$ and $\psi()$ closely off the formulas in [77].

Finally, the valence score $\hat{Y}^k$ for window X^k is predicted using a simple feed-forward network:

$$\hat{Y}^k = \Phi\left(\boldsymbol{H}^k; \boldsymbol{W}_\Phi\right) \tag{6}$$

where Φ is a multi-layer perceptron (MLP) which includes three linear layers using *ReLU* as activation function, and $\boldsymbol{W}_\Phi$ are parameters of the MLP.

Training WBA. WBA is trained by minimizing the *mean squared error* loss between the predicted scores $\hat{Y}$ and the ground-truth scores Y:

$$\arg\min_{\boldsymbol{W}} \frac{1}{n} \sum_{k=1}^{n} (\hat{Y}^k - Y^k)^2 \tag{7}$$

where $\boldsymbol{W} = \{\boldsymbol{W}_\psi, \boldsymbol{v}_a, \boldsymbol{W}_a, \boldsymbol{W}_\Psi, \boldsymbol{W}_\Phi\}$ are model's parameters to be learnt. The ground-truth Y^k for window k is the averaged valence across the window.

4. Experiment

4.1. Dataset

To evaluate our method, we used the Stanford Emotional Narratives Dataset (SEND) [45][1], a recently released dataset consisting of videos of people narrating emotional events in their lives. The dataset consists of 193 video clips (i.e. stories), lasting an average of 2 minutes and 15 seconds. For our purposes, we only use the text modality of their multimodal dataset. The *training*, *validation* and *test* sets have 114, 40 and 39 stories respectively, using the same splits as in [45].

The stories were annotated by an average of 20 independent raters, who used a continuous slider to rate emotional valence: how they thought the person in the video was feeling as they were telling the story. This provided a continuous rating (rescaled to -1 to 1) that was sampled every 0.5 second. The gold-standard label is the Evaluator Weighted Estimator (EWE) of the independent valence ratings, which is a weighted average that downweights idiosyncratic outliers and helps to ensure more reliable ratings. We refer the reader to the original dataset paper for more details on the dataset collection and annotation.

4.2. Experiment Setup

Pre-trained word embedding models. We investigate the explainability of our window-based attention model when using different pre-trained word embedding models including both *context-free* and *contextual* embeddings. Context-free word embeddings contain a single word representation (i.e. embedding) for each word in a pre-defined vocabulary. By contrast, contextual word embeddings generate a representation for a word that depends on the context the word appears. For context-free word embeddings, we use GloVe [49] (840B tokens, 300d vectors). For contextual word embeddings, we use BERT [15], DistilBERT [53], RoBERTa [37], GPT-2 [51], and ELECTRA [14]. To obtain the contextual embedding for a word, we use the corresponding last hidden state when the sentence (or window) containing the word is given as input.

Our methods. For our WBA model, we choose window sizes of 3, 5, or 10 seconds (denoted as WBA-3s, WBA-5s, and WBA-10s respectively). For each window size, we choose the best padding length based on validation set, the values are 15, 25 and 40 for window size of 3, 5, and 10 seconds, respectively. We first compare the effect of using these different window sizes and then use the best setting for the rest of the experiments.

During *training*, the target prediction for each window is the averaged valence ratings across the window. During *prediction*, the predicted valence for a window is repeated based on the corresponding timing of the window.

Baselines. We compare with previously published models on the SEND including Encoder-Decoder LSTM (EncDec LSTM) [45], Variational Recurrent Neural Network (VRNN) [45], Simple Fusion Transformer (SFT) [73], Memory Fusion Transformer (MFT) [73], Attention-based Encoder-Decoder (Att-ED) [71], and Affect2MM [43].

[1] `https://github.com/StanfordSocialNeuroscienceLab/SEND`

EncDec LSTM consists of two LSTM layers with a local attention layer in between. VRNN takes into account implicit sources of variation by adding a generative component to an RNN. SFT uses CNNs to process input features before passing to a Transformer layer and predicts a valence score for each window using a LSTM decoder layer. MFT adapts Memory Fusion Network [76] to learn attention across different modalities. Att-ED adopts the encoder-decoder architecture in which the encoder is identical to Transformer's encoder and the decoder consists of a LSTM, followed by a multilayer perceptron for predicting valence scores. Affect2MM uses attention-based methods and Granger causality to model the temporal causality for predicting emotional state evoked in videos. In Table 2, we report the best result for each baseline model, i.e., EncDec LSTM, VRNN, SFT and Att-ED on text modality, MFT on visual, audio and text modalities, and Affect2MM on audio and text modalities (i.e., Affect2MM-AT).

To evaluate the effect of using our proposed two-level LSTM with attention, we implement several additional baselines as "lesioned" versions of our model: flattened LSTM (F-LSTM) and window-based without attention models (WB). F-LSTM has only one bidirectional LSTM and predicts valence for each word using the same output layer's structure as in WBA. WB is the same as WBA except that in WB, the window encoding is the last hidden state of the local-level LSTM. We run WB for window size of 3, 5, and 10 seconds that are denoted as WB-3s, WB-5s and WB-10s, respectively.

Settings and Hyperparameters. For both local-level and global-level LSTMs, we use bidirectional LSTMs with hidden dimension of 128 ($d = D = 128$). The output dimensions of the linear layers of the output MLP Φ are 128, 64 and 1, respectively. In WBA model, there are about 943K trainable parameters in total. We used the Adam optimizer [27] with a learning rate of 0.001. During training, we apply a dropout rate of 1% to the input embeddings. All the experiments we conducted (including F-LSTM, WB, WBA) were run $n = 20$ times and we report the average scores.

Evaluation metric. We use the same evaluation metric as in [45]: the Concordance Correlation Coefficient (CCC) between the model's predicted valence $\hat{Y}$ and the gold-standard label Y. The CCC [32] is given by:

$$CCC \equiv \frac{2\rho_{Y\hat{Y}}\sigma_Y\sigma_{\hat{Y}}}{\sigma_Y^2 + \sigma_{\hat{Y}}^2 + (\mu_Y - \mu_{\hat{Y}})^2} \tag{8}$$

where $\rho_{Y\hat{Y}}$ gives the Pearson correlation of Y and $\hat{Y}$, and $\{\mu_Y, \mu_{\hat{Y}}\}$ and $\{\sigma_Y, \sigma_{\hat{Y}}\}$ give the means and standard deviations of Y, $\hat{Y}$ respectively.

4.3. Evaluating the window-based attention model.

We first evaluate the effects of using the hierarchical (two-level) LSTM and the attention mechanism by comparing F-LSTM, WB and WBA models. We use GloVe embedding for this experiment. Each model is run n times ($n = 20$) to ensure stable results. We compute and report the mean CCC and the standard deviation across the runs for the validation and test sets. Table 1a shows the results of our proposed models and the baselines. Our baseline window-based models without attention (WB) outperform F-LSTM, showing that a hierarchical LSTM performs better at encoding sequential information for long text content. Attention further improves performance:

Model (GloVe)		Validation	Test
No Att.	F-LSTM	0.42 (0.06)	0.43 (0.07)
	WB-3s	0.51 (0.02)	0.50 (0.02)
	WB-5s	0.48 (0.02)	0.45 (0.02)
	WB-10s	0.41 (0.01)	0.41 (0.03)
With Att.	WBA-3s	0.55 (0.02)	0.55 (0.02)
	WBA-5s	**0.57** (0.02)	**0.58** (0.01)
	WBA-10s	0.55 (0.02)	0.52 (0.02)

(a) Evaluating the effect of window-based (WB) approach and the use of attention (Att.). Window lengths are in 3, 5, and 10 seconds denoted as 3s, 5s, and 10s, respectively. In general, WB models outperform flattened LSTM (F-LSTM). Using attention (With Att.) improves the results and the setting of 5-second window with attention (WBA-5s) achieves the best results. GloVe embedding is used.

WordEmb	Sentence	WBA-5s
GloVe	0.54 (0.03)	0.58 (0.01)
BERT	0.57 (0.03)	**0.65** (0.02)
DistilBERT	0.53 (0.02)	0.63 (0.02)
RoBERTa	0.61 (0.02)	0.62 (0.02)
GPT-2	0.42 (0.03)	0.40 (0.03)
ELECTRA	0.36 (0.02)	0.38 (0.02)

(b) Evaluating our WBA model with different pre-trained word embedding models (WordEmb). An input window is a sentence or a window of 5 seconds (WBA-5s). Average CCC scores for the test set are reported. Using window of 5 seconds outperforms using sentence for almost all the word embeddings (except for GPT-2). WBA-5s with BERT embedding achieves the best result.

Table 1: Evaluating window-based attention (WBA) model in predicting emotion valence. Concordance correlation coefficient (CCC) results of different setups of the model. Each result is averaged over 20 runs. All models are consistent over different runs as the standard deviations of the results (shown in parentheses) are very low.

The WBA model outperforms WB across all the settings. Interestingly, with the attention module, WBA better handles longer windows as the best window size option for WBA is 5 seconds, whereas in WB, it is 3 seconds.

Next, we sought to compare different word embeddings by comparing GloVe with other contextual word embedding models (Table 1b). Since contextual word embedding models were trained using whole sentences, we compare the model performances when our window input is sentence-based (i.e. each sentence is a window) compared to time-based (using a 5-second window, which was the best setting for WBA). Table 1b shows the CCC scores for the test set for both sentence-based (Sentence) and time-based (WBA-5s). WBA-5s performs better than sentence-based in almost all the word embeddings, except for GPT-2. This could be due to the nature of the dataset we used, which was created based on spoken narrative where the content and sentences (e.g., with occasional disfluencies) are not as standard as in written linguistic data. Contextual word embeddings such as BERT, DistilBERT and RoBERTa performs better than the context-free GloVe, but we also find that GPT-2 and ELECTRA do not perform as well as GloVe. The low standard deviation values imply that the models are consistent over the different runs. Overall, WBA-5s with BERT embedding does the best.

Third, we compare the performance of our WBA-5s-BERT model with previously published results on this dataset. Table 2 shows the comparison results for validation and test sets. Except for Affect2MM-AT, our model outperforms all the other baselines with significant margins on both validation and test sets. Compared to Affect2MM-AT, the performance on validation set is comparable and WBA-5s-BERT performs better on the test set (note that our model only uses text modality, whereas Affect2MM-AT uses audio and text modalities). It is also noteworthy that the metric, CCC, ranges from -1 to 1 and the averaged human performance on the test set is only 0.50, implying that achieving high CCC for this task is not easy. Despite this, our WBA-5s model using BERT significantly outperforms human performance with a CCC of 0.65 on the test set (two-tailed paired t-test, t=3.81, $p<$ .001). In the next section, we analyze the input

Model		Validation	Test
Previously Published	EncDec LSTM [45]	0.38	0.40
	VRNN [45]	0.43	0.42
	SFT [73]	0.34	0.34
	MFT [73]	0.42	0.44
	Att-ED [71]	-	0.54
	Affect2MM-AT [43]	**0.59**	0.60
Our Model	WBA-5s-BERT	0.54	**0.65**
Human Performance [45]		0.47	0.50

Table 2: Averaged concordance correlation coefficient (CCC) scores for the previously published baselines and our proposed model. Our window-based model with window size of 5 seconds using BERT (WBA-5s-BERT) performs the best among all the baselines on the test set, even outperforming average human performance.

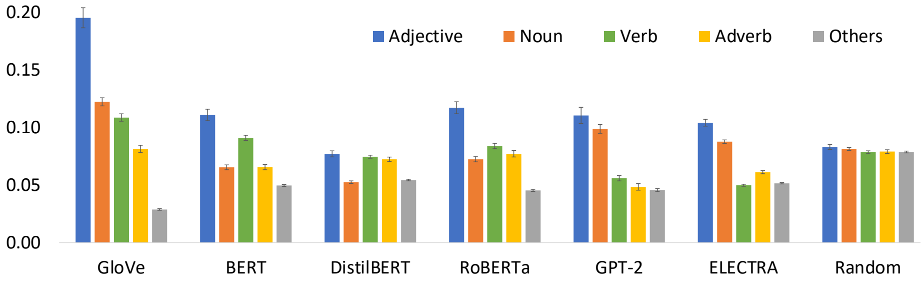


Figure 2: Average attention weights for part-of-speech (POS) tags with standard errors. The right-most column shows the results for the case where attention weights are randomly distributed within the window, showing that the distribution across POS tags is approximately uniform. We find that across all the models tested, Adjectives receive the most attention weight.

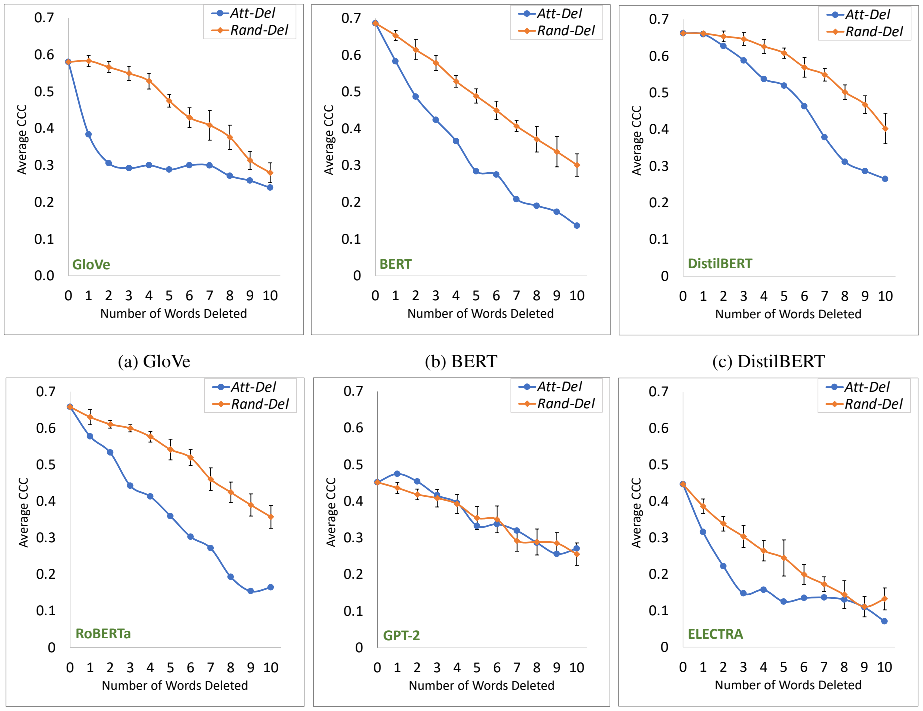
tokens assigned high attention weights by our WBA-5s model in terms of their *syntax* and *semantics*.

4.4. Attention Analysis

In order to qualitatively evaluate our proposed WBA model, we analyzed the attention weights generated by the best model on unseen data, i.e., the test set. We choose the best performing model (WBA-5s) and examined the words (tokens) that received the most attention from the model, in terms of syntactic and semantic information. We investigated the attention weights for GloVe, BERT, DistilBERT, RoBERTa, GPT-2 and ELECTRA. For each word embedding, we used the best model across all the different runs to generate the analysis results.

4.4.1. Analysing Syntax

The key pieces of information to understanding the emotions in a narrative life story are the various events described in the story. Syntactic information such as part-of-speech (POS) has been widely used for event extraction [70], which often proceeds by extracting text relations mediated by Verbs and Verb patterns [20, 19], and relations mediated by Nouns and Adjectives [54]. Emotions in particular are reactions caused by



(a) GloVe

(b) BERT

(c) DistilBERT

(d) RoBERTa

(e) GPT-2

(f) ELECTRA

Figure 3: Average concordance correlation coefficient (CCC) when randomly deleting words (*Rand-Del*) or deleting words with highest attention weights (*Att-Del*). *Rand-Del*'s results are averaged over 10 runs of randomly deleting words. Standard deviations across the runs are also visualized.

events [18], and therefore, we hypothesize that to understand the emotions in a story, the model should attend more to Verbs, Nouns and Adjectives. In this section, we examine this hypothesis by analyzing the attention weights distributed to POS tags.

We ran the Stanford CoreNLP POS Tagger [61] on stories' text content to get the POS tag for each word and assign the word's attention weight to the corresponding tag. We are interested in the coarse-grained POS tags, so we grouped tags of the same type—e.g. adjective comparative and superlative are grouped into adjective—and computed the average attention weights and the standard errors. The results are plotted in Figure 2. To serve as a comparison to understand our results better, we also included a "*Random*" subgraph where words are randomly assigned attention weights (normalized for each window). As we would expect, for randomly-assigned weights, the attention distribution across the POS-tags is uniform. Interestingly, our WBA-5s model attended the most to Adjective for all the different word embeddings tested. Overall, Nouns and Verbs also received high attention scores across all the models, but their relative importance differed by model. This result supports our hypothesis that the model should attend more on Verbs, Nouns and Adjectives.

4.4.2. Analysing Semantics

Word deletion. To examine the importance of highly attended words, we ran a *word deletion* experiment in which we delete k words in the input of each window. We compare the results when we randomly delete k words (*Rand-Del*) versus deleting the k words that have the highest attention weights in a window (*Att-Del*). For each value of k, we ran 10 times for *Rand-Del* and report the average result with its standard deviation across the runs. We varied the number of words to be deleted from k =1 to 10, and a window is ignored if it has less than k words. The goal of this experiment is to evaluate the impact of highly-attended words in a window to the model's output. We conducted this experiment for all six pre-trained word embedding models.

As shown in Figure 3, except for GPT-2, deleting highly attended words drastically reduced the models' performance, compared to deleting random words. The largest drop occured for GloVe: When using GloVe, deleting the most-attended word in a window already significantly reduced performance (from 0.60 to 0.38; paired t-test, t=3.39 p=.002). The performance continued to drop significantly when deleting two mostly attended words in a window (from 0.38 to 0.31; paired t-test, t=2.83 p=0.007). From $k = 3$ onwards, *Att-Del*'s performance fluctuated before decreasing. The differences between *Att-Del* and *Rand-Del* are also significant with $p < .05$ for $k = 1...5$. For BERT, when deleting the most-attended word in a window, the performance also drops significantly from 0.69 to 0.58 (paired t-test, t=4.40, $p < .001$)

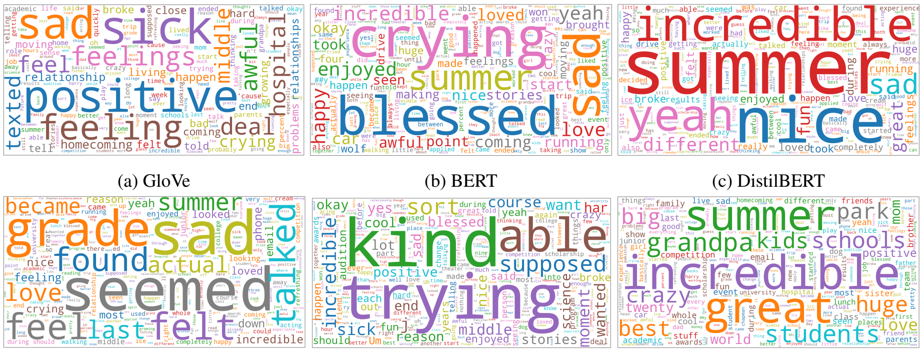
It is interesting to note that even though ELECTRA does not do as well on this task (average test-CCC of 0.38; Table 1), it still shows the same consistent pattern where there is a significant difference between *Att-Del* and *Rand-Del*. The only exception to this pattern is GPT-2, where the drop in performance is similar across both *Att-Del* and *Rand-Del* (and we note that it is also *not* among the top performing word embeddings, from Table 1). We do not know why this might be so, but we hypothesize that perhaps the performance of GPT-2 on this task might come more from other parts of the model (e.g., word embedding weights) rather than the attention. Future work should investigate this on other tasks.

We concatenated the results from *Att-Del* and *Rand-Del* for all the values of k and perform the paired t-test for each word embedding. The results show that the differences between *Att-Del* and *Rand-Del* for all the word embeddings, except for GPT-2, are significant (with p< .001). The results clearly show that the top words attended by our model (for all the word embeddings except for GPT-2) have a significant impact on the model performance, providing evidence that the attention weights tend to pick out words that are important for model performance.

Word cloud. Figure 4 shows the word clouds generated from the average attention weight of each word in the test set, for each of the word embeddings studied. We can see that the most-attended words (e.g. "positive", "sick", "blessed", "crying", "summer") are important to understand the emotions in a story. Again, the only exception seems to be GPT-2, where the most-attended to words do not seem to carry as much emotional meaning as the rest of the word embeddings.

4.4.3. Case study

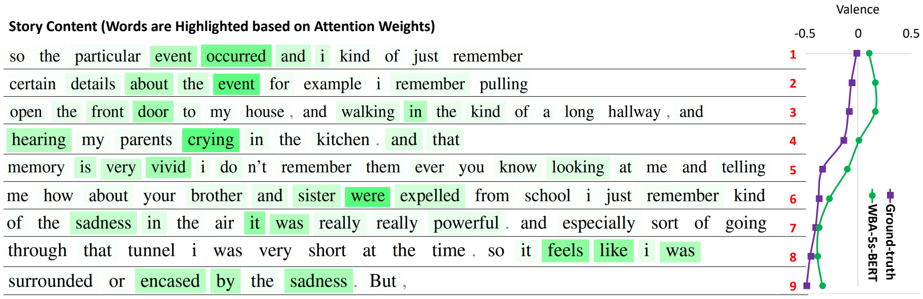
In this section, we show qualitative results generated by our model using the best settings, i.e. WBA-5s using BERT.



(a) GloVe (b) BERT (c) DistilBERT

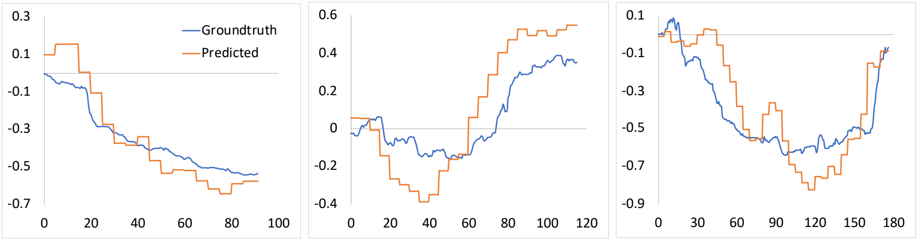
(d) RoBERTa (e) GPT-2 (f) ELECTRA

Figure 4: Word cloud based on the average attention weight of each word in the test set. We only include words appearing at least 5 times. Popped-out words in GloVe and BERT are important to semantically understand a story. Highly attended words in GPT-2, however, do not carry emotional meaning which explains the not-good performance and the behaviour in delete-word experiment.


Story Content (Words are Highlighted based on Attention Weights)

1 so the particular event occurred and i kind of just remember
2 certain details about the event for example i remember pulling
3 open the front door to my house , and walking in the kind of a long hallway , and
4 hearing my parents crying in the kitchen . and that
5 memory is very vivid i do n't remember them ever you know looking at me and telling
6 me how about your brother and sister were expelled from school i just remember kind
7 of the sadness in the air it was really really powerful . and especially sort of going
8 through that tunnel i was very short at the time . so it feels like i was
9 surrounded or encased by the sadness . But ,

Figure 5: Visualization of attended words in a window. The stronger the highlighted color, the higher attention weight received by the word. Gray words indicate words removed in preprocessing. Right: The corresponding ground-truth and model-predicted valence of each window, transposed such that time goes vertically downwards, and positive valence is towards the right.



(a) Story 1 (b) Story 2 (c) Story 3

Figure 6: Predictions of our best model, WBA-5s-BERT, compared with ground-truth valences for three stories in the test set. The x-axis is time (in seconds) and the y-axis is the valence ranging from -1 (very negative) to 1 (very positive). Our model is able to predict the valence-trends as the stories go on.

Attention Visualization. To visualize the attended words in a window, we highlight words based on the attention weights they received, as shown in Figure 5. The figure includes the first 9 windows ($\sim$45s) of a story in the test set. The stronger the highlighted color, the higher attention weight received by the word. The corresponding ground-truth and predicted valences for each window are also plotted on the right. As shown in the Figure, our WBA-5s model using BERT is able to attend to important words when predicting the desired valence. In the first 3 windows, there is not enough information for the model to make the decision whether the story is positive or negative; But at window 4, it attended to 'hearing' and 'crying', and this was accompanied by a decrease in the predicted valence. In window 6, our model attended to more semantically meaningful words: '... sister were expelled', and continued to predict a negative valence score. In the later windows, the model attended on emotional words including 'sadness', 'powerful', 'feels', 'encased by ... sadness'.

Visualising Predictions. Figure 6 shows the predictions of our proposed model along with the ground-truth valences for three stories in the test set. Figure 6a visualizes the entire story from Figure 5. After the excerpt shown in Figure 5, the story continues with how sad the father was, and this was accompanied by a decreasing valence score. Story 2 (Figure 6b) was about a student who had just finished high school and was looking at college. The tuition fees were high and their family had financial concerns, making it a difficult decision. The turning point in the valence, again captured by both ground-truth and our model predictions, occurred when the student won a full scholarship. As the story went on with how happy the student was, the model predicted a positive turn in valence. Story 3 (Figure 6c) was about a junior high school student who was in a running team and took part in parties. Others found out about it and the student felt embarrassed. Our model was able to generate a downward trend for the valence of the story. Interestingly, the model was able to reverse the trend (similar to the ground-truth) when the narrator mentioned that because of the incident, they started to focus more on studies and hence partied less. Our model's predictions closely followed the ground-truth emotion trends of the story even when the story tended to be all negative.

5. Discussion and Conclusions

In this paper, we proposed a *Window-Based Attention* (WBA) model combining a hierarchical LSTM with attention mechanism, for an emotion understanding task that predicts real-valued emotion valence of a time-series narrative. Experimental results using different word embeddings on a naturalistic emotion narratives dataset show that our WBA model outperforms state-of-the-art models, even surpasses averaged human performance (when using GloVe, BERT, DistilBERT and RoBERTa).

Our second main contribution is a set of extensive analyses on the attention weights that together provide evidence that the attention layer is capable of attending to words that carry emotional semantic meaning (conditioned on the model reaching human-level performance). We should point out that although several analyses (e.g., Fig. 5 and Fig. 6) focus on selected test set examples, they are meant to be illustrative, and the rest of our analyses (e.g., word deletion, POS tags) are done on the full test set. Indeed, the analyses should be viewed together as a collection of evidence towards the idea that attention does uncover task-relevant semantics. Also relevant to this discussion is some recent work where we examined self-attention weights in a different model, the Transformer with GloVe word embeddings [71]; We found that attention scores are correlated with an external source of emotional semantic meaning, labels from emotion lexicons.

In this work, we present the analyses for one attention mechanism, i.e., *softmax-concat* (or *additive*) [38]. Similar evaluations and analyses can be applied for other attention mechanisms such as *softmax-dot* [38], *content-based attention* [21] and *scaled dot-product attention* [63]. Several interesting research questions are: 1) "How do the different attention mechanisms influence model performance on a given task?"; 2) "Can all the different attention mechanisms uncover semantic information?"; and 3) "What factors affect the ability of an attention mechanism in uncovering semantic information?". Having comparisons between different mechanisms will shed more light on the debate. We leave this direction to the future work.

We chose an emotion understanding task specifically because it contains intuitive semantics, but we see no reason why, on other tasks, one could not also design similar analyses to probe the link between attention and semantics. Future work should extend similar analyses to other NLP tasks. For example, our model "paid more attention" to Adjectives, then Nouns and Verbs, then the other part-of-speech types: This pattern could very well depend on the task at hand.

In sum, our paper adds to the debate on how one can 'interpret' attention. Our work supports previous work that attention models encode syntax [60] and human-judged importance [46], by showing evidence that attention-based models may learn to place higher attention weights on words that humans find semantically meaningful to the downstream task.

Acknowledgements

This work was supported by a Singapore Ministry of Education Academic Research Fund Tier 1 grant to DCO, and by A*STAR under its Knowledge Extraction, Modelling, and Explainable Reasoning for General Expertise programme (Award A19E2b0098). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of A*STAR.

References

[1] Md Shad Akhtar, Asif Ekbal, and Erik Cambria. 2020. How Intense Are You? Predicting Intensities of Emotions and Sentiments Using Stacked Ensemble. *IEEE Computational Intelligence Magazine*, 15(1):64–75.

[2] Leila Arras, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. 2017. Explaining Recurrent Neural Network Predictions in Sentiment Analysis. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 159–168.

[3] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. 2020. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges Toward Responsible AI. *Information Fusion*, 58:82–115.

[4] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representations, ICLR 2015*.

[5] Mohammad Ehsan Basiri, Shahla Nemati, Moloud Abdar, Erik Cambria, and U Rajendra Acharya. 2021. ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for Sentiment Analysis. *Future Generation Computer Systems*, 115:279–294.

[6] Swapnil Bhosale, Rupayan Chakraborty, and Sunil Kumar Kopparapu. 2020. Deep Encoded Linguistic and Acoustic Cues for Attention Based End to End Speech Emotion Recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7189–7193. IEEE.

[7] Rafael A Calvo and Sidney D'Mello. 2010. Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications. *IEEE Transactions on Affective Computing*, 1(1):18–37.

[8] Erik Cambria, Soujanya Poria, Alexander Gelbukh, and Mike Thelwall. 2017. Sentiment Analysis Is a Big Suitcase. *IEEE Intelligent Systems*, 32(6):74–80.

[9] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964. IEEE.

[10] Danqi Chen, Jason Bolton, and Christopher D Manning. 2016. A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 2358–2367.

[11] Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long Short-term Memory-networks for Machine Reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561.

[12] Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive Sentence Summarization with Attentive Recurrent Neural Networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

[13] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What Does BERT Look at? An Analysis of BERT's Attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.

[14] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training Text Encoders as Discriminators Rather than Generators. *arXiv preprint arXiv:2003.10555*.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

[16] Li Dong and Mirella Lapata. 2016. Language to Logical Form with Neural Attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 33–43.

[17] Tim Donkers, Timm Kleemann, and Jürgen Ziegler. 2020. Explaining Recommendations by Means of Aspect-based Transparent Memories. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pages 166–176.

[18] Phoebe C Ellsworth and Klaus R Scherer. 2003. *Appraisal Processes in Emotion.*, volume 572. Oxford University Press.

[19] Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam Mausam. 2011. Open Information Extraction: The Second Generation. In *Twenty-Second International Joint Conference on Artificial Intelligence*, volume 11, pages 3–10.

[20] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying Relations for Open Information Extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545.

[21] Alex Graves, Greg Wayne, and Ivo Danihelka. 2014. Neural turing machines. *arXiv preprint arXiv:1410.5401*.

[22] Christopher Grimsley, Elijah Mayfield, and Julia RS Bursten. 2020. Why Attention is Not Explanation: Surgical Intervention and Causal Reasoning about Neural Models. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 1780–1790.

[23] Yaru Hao, Li Dong, Furu Wei, and Ke Xu. 2020. Self-Attention Attribution: Interpreting Information Interactions Inside Transformer. *arXiv preprint arXiv:2004.11207*.

[24] John Hewitt and Christopher D Manning. 2019. A Structural Probe for Finding Syntax in Word Representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138.

[25] Feiran Huang, Kaimin Wei, Jian Weng, and Zhoujun Li. 2020. Attention-based Modality-gated Networks for Image-text Sentiment Analysis. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(3):1–19.

[26] Sarthak Jain and Byron C Wallace. 2019. Attention Is Not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3543–3556.

[27] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015*.

[28] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention Is Not Only a Weight: Analyzing Transformers with Vector Norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075.

[29] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. 2017. A Hierarchical Approach for Generating Descriptive Image Paragraphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 317–325.

[30] Ayush Kumar and Jithendra Vepa. 2020. Gated Mechanism for Attention Based Multi Modal Sentiment Analysis. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4477–4481. IEEE.

[31] Zheng Lian, Jianhua Tao, Bin Liu, and Jian Huang. 2019. Conversational Emotion Analysis via Attention Mechanisms. *arXiv preprint arXiv:1910.11263*.

[32] Lawrence I-Kuei Lin. 1989. A Concordance Correlation Coefficient to Evaluate Reproducibility. *Biometrics*, pages 255–268.

[33] Rui Lin, Shujie Liu, Muyun Yang, Mu Li, Ming Zhou, and Sheng Li. 2015. Hierarchical Recurrent Neural Network for Document Modeling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 899–907.

[34] Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open Sesame: Getting inside BERT's Linguistic Knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253.

[35] Ning Liu, Bo Shen, Zhenjiang Zhang, Zhiyuan Zhang, and Kun Mi. 2019. Attention-based Sentiment Reasoner for Aspect-based Sentiment Analysis. *Human-centric Computing and Information Sciences*, 9(1):1–17.

[36] Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content Attention Model for Aspect Based Sentiment Analysis. In *Proceedings of the 2018 World Wide Web Conference*, pages 1023–1032.

[37] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A Robustly Optimized Bert Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

[38] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

[39] Yukun Ma, Haiyun Peng, and Erik Cambria. 2018. Targeted Aspect-based Sentiment Analysis via Embedding Commonsense Knowledge into an Attentive LSTM. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[40] Andre Martins and Ramon Astudillo. 2016. From Softmax to Sparsemax: A Sparse Model of Attention and Multi-label Classification. In *International Conference on Machine Learning*, pages 1614–1623.

[41] Seyedmahdad Mirsamadi, Emad Barsoum, and Cha Zhang. 2017. Automatic Speech Emotion Recognition Using Recurrent Neural Networks with Local Attention. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2227–2231. IEEE.

[42] Trisha Mittal, Pooja Guhan, Uttaran Bhattacharya, Rohan Chandra, Aniket Bera, and Dinesh Manocha. 2020. EmotiCon: Context-Aware Multimodal Emotion Recognition Using Frege's Principle. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14234–14243.

[43] Trisha Mittal, Puneet Mathur, Aniket Bera, and Dinesh Manocha. 2021. Affect2MM: Affective Analysis of Multimedia Content Using Emotion Causality. *arXiv preprint arXiv:2103.06541*.

[44] Pooya Moradi, Nishant Kambhatla, and Anoop Sarkar. 2019. Interrogating the Explanatory Power of Attention in Neural Machine Translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 221–230.

[45] Desmond Ong, Zhengxuan Wu, Zhi-Xuan Tan, Marianne Reddan, Isabella Kahhale, Alison Mattek, and Jamil Zaki. 2019. Modeling Emotion in Complex Stories: The Stanford Emotional Narratives Dataset. *IEEE Transactions on Affective Computing*.

[46] Nikolaos Pappas and Andrei Popescu-Belis. 2016. Human Versus Machine Attention in Document Classification: A Dataset with Crowdsourced Annotations. In *Proceedings of The Fourth International Workshop on Natural Language Processing for Social Media*, pages 94–100.

[47] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the Difficulty of Training Recurrent Neural Networks. In *International Conference on Machine Learning*, pages 1310–1318. PMLR.

[48] Zhichao Peng, Xingfeng Li, Zhi Zhu, Masashi Unoki, Jianwu Dang, and Masato Akagi. 2020. Speech Emotion Recognition Using 3d Convolutions and Attention-based Sliding Recurrent Networks with Auditory Front-ends. *IEEE Access*, 8:16560–16572.

[49] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing*, pages 1532–1543.

[50] Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C Lipton. 2020. Learning to Deceive with Attention-Based Explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4782–4793.

[51] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models Are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9.

[52] Waleed Ragheb, Jérôme Azé, Sandra Bringay, and Maximilien Servajean. 2019. Attention-based Modeling for Emotion Detection and Classification in Textual Conversations. *arXiv preprint arXiv:1906.07020*.

[53] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, A Distilled Version of Bert: Smaller, Faster, Cheaper and Lighter. *arXiv preprint arXiv:1910.01108*.

[54] Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. 2012. Open Language Learning for Information Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics.

[55] Sofia Serrano and Noah A Smith. 2019. Is Attention Interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951.

[56] Raksha Sharma, Mohit Gupta, Astha Agarwal, and Pushpak Bhattacharyya. 2015. Adjective Intensity and Sentiment Analysis. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2520–2526.

[57] Man-Chin Sun, Shih-Huan Hsu, Min-Chun Yang, and Jen-Hsien Chien. 2018. Context-aware Cascade Attention-based RNN for Video Emotion Recognition. In *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pages 1–6. IEEE.

[58] Xiaobing Sun and Wei Lu. 2020. Understanding Attention for Text Classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3418–3428.

[59] Lorenzo Tarantino, Philip N Garner, and Alexandros Lazaridis. 2019. Self-Attention for Speech Emotion Recognition. In *Interspeech*, pages 2578–2582.

[60] Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT Rediscovers the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601.

[61] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich Part-of-speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Conference of the North American chapter of the Association for Computational Linguistics on Human Language Technology-volume 1*, pages 173–180. Association for Computational Linguistics.

[62] Shikhar Vashishth, Shyam Upadhyay, Gaurav Singh Tomar, and Manaal Faruqui. 2019. Attention Interpretability Across NLP Tasks. *arXiv preprint arXiv:1909.11218*.

[63] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

[64] Jesse Vig and Yonatan Belinkov. 2019. Analyzing the Structure of Attention in a Transformer Language Model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76.

[65] Elena Voita, Rico Sennrich, and Ivan Titov. 2020. Analyzing the Source and Target Contributions to Predictions in Neural Machine Translation. *arXiv preprint arXiv:2010.10907*.

[66] Jin Wang, Bo Peng, and Xuejie Zhang. 2018. Using a Stacked Residual LSTM Model for Sentiment Intensity Prediction. *Neurocomputing*, 322:93–101.

[67] Yanan Wang, Jianming Wu, and Keiichiro Hoashi. 2019. Multi-attention Fusion Network for Video-based Emotion Recognition. In *2019 International Conference on Multimodal Interaction*, pages 595–601.

[68] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based Lstm for Aspect-level Sentiment Classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

[69] Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not Explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20.

[70] Fei Wu and Daniel S Weld. 2010. Open Information Extraction Using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127. Association for Computational Linguistics.

[71] Zhengxuan Wu, Thanh-Son Nguyen, and Desmond Ong. 2020. Structured Self-Attention Weights Encodes Semantics in Sentiment Analysis. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 255–264.

[72] Zhengxuan Wu and Desmond C Ong. 2021. Context-Guided BERT for Targeted Aspect-Based Sentiment Analysis. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*.

[73] Zhengxuan Wu, Xiyu Zhang, Tan Zhi-Xuan, Jamil Zaki, and Desmond C Ong. 2019. Attending to Emotional Narratives. In *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 648–654. IEEE.

[74] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

[75] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2016. Video Paragraph Captioning Using Hierarchical Recurrent Neural Networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4584–4593.

[76] Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Memory Fusion Network for Multi-view Sequential Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[77] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent Neural Network Regularization. *arXiv preprint arXiv:1409.2329*.

[78] Yuanyuan Zhang, Jun Du, Zirui Wang, Jianshu Zhang, and Yanhui Tu. 2018. Attention Based Fully Convolutional Network for Speech Emotion Recognition. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1771–1775. IEEE.

[79] Zixing Zhang, Jing Han, Eduardo Coutinho, and Björn Schuller. 2018. Dynamic Difficulty Awareness Training for Continuous Emotion Prediction. *IEEE Transactions on Multimedia*, 21(5):1289–1301.

[80] Zufan Zhang, Yang Zou, and Chenquan Gan. 2018. Textual Sentiment Analysis via Three Different Attention Convolutional Neural Networks and Cross-modality Consistent Regression. *Neurocomputing*, 275:1407–1415.