# Understanding Neural Networks through Representation Erasure

**Jiwei Li, Will Monroe and Dan Jurafsky**
Computer Science Department, Stanford University, Stanford, CA, USA
jiweil,wmonroe4,jurafsky@stanford.edu



## Abstract

While neural networks have been successfully applied to many natural language processing tasks, they come at the cost of interpretability. In this paper, we propose a general methodology to analyze and interpret decisions from a neural model by observing the effects on the model of erasing various parts of the representation, such as input word-vector dimensions, intermediate hidden units, or input words. We present several approaches to analyzing the effects of such erasure, from computing the relative difference in evaluation metrics, to using reinforcement learning to erase the minimum set of input words in order to flip a neural model's decision. In a comprehensive analysis of multiple NLP tasks, including linguistic feature classification, sentence-level sentiment analysis, and document level sentiment aspect prediction, we show that the proposed methodology not only offers clear explanations about neural model decisions, but also provides a way to conduct error analysis on neural models.

## 1 Introduction

An long-standing criticism of neural network models is their lack of interpretability. Unlike traditional feature-based models that optimize weights on human interpretable features, neural network models operate like a black box: using vector representations (as opposed to human-interpretable features) to represent text units in the input, applying multiple layers of non-linear transformations, and outputting a scalar representing the decision. Mystery exists at all levels of a neural model: At input layers, what does each word vector dimension stand for? What do hidden units in intermediate levels stand for? How does the model combine meaning from different parts of the sentence, filtering the informational wheat from the chaff? How is the final decision made at the output layer?

In this paper, we propose a general methodology for interpreting neural network behavior by analyzing the effect of erasing pieces of the representation, to see how such changes affect a neural model's decisions. This representation erasure can be performed on various levels of representation, including input word-vector dimensions, input words or phrases, and intermediate hidden units. We apply algorithms of varying complexity for performing this erasure and analyzing the output. At the simplest end, one can directly compute the difference in log likelihood on gold-standard labels when representations are erased; on the more sophisticated end, one can utilize a reinforcement learning model to find the minimal set of words that must be erased to change the model's decision.

We observe that the proposed framework offers many interpretable explanations for various aspects of neural models: (1) how a neural model picks word-vector dimensions for linguistic feature classification (parts of speech, named entity recognition, chunking, etc.); (2) how neural models select and filter important words, phrases, and sentences in sentiment analysis; (3) why architectures like long short-term memory networks (LSTMs) perform more competitively than standard recurrent neural networks (RNNs); and most importantly (4) why neural models make mistakes (error analysis).

## 2 Related Work

Efforts to understand neural vector space models in natural language processing (NLP) include visualizing state activation (Hermans and Schrauwen, 2013; Karpathy et al., 2015), interpreting semantic dimensions by presenting human users with a list of words and ask them to choose the one that does not belong to the list (Fyshe et al., 2015; Murphy et al., 2012), linking dimensions with semantic lexicons or word properties (Faruqui et al., 2014; Herbelot and Vecchi, 2015), learning sparse interpretable word vectors (Faruqui et al., 2015), and empirical studies of LSTM components (Greff et al., 2015; Chung et al., 2014).

Each of these approaches successfully reveals a particular aspect of neural network decisions that is necessary for understanding, but each is also constrained by the scope of its applicability. Karpathy et al. (2015) visualize the neural generation models from an error-analysis point of view, by analyzing predictions and errors from a recurrent neural models. The approach shows the intriguing dynamics of hidden cells in LSTMs but is limited to a few manually-inspected cases such as brace opening and closing. Li et al. (2015) use the first-order derivative to examine the saliency of input features, but they rely on the overly strong assumption that the decision score is a linear combination of input features.

Other closely related work includes that of Lei et al. (2016), who train a separate generator that extracts a subset of text which lead to a similar decision to the original input to form an interpretable summary. Shi et al. (2016) study the role of vector dimensions in sequence generation tasks; they successfully detect vector dimensions that perfectly track length in a toy sequence generation problem, but not in full-scale neural machine translation (MT). Strobelt et al. (2016) develop an interactive system that allows users to select LSTM intermediate states and align these state changes to domain specific structural annotations. The structure of attention-based models (Bahdanau et al., 2014; Luong et al., 2015; Sukhbaatar et al., 2015; Rush et al., 2015; Xu and Saenko, 2015) enables another way of explaining the workings of neural models. Attention models are specifically designed to handle tasks in which an alignment needs to be modeled between inputs and outputs, like machine translation. The proposed framework is an analysis method rather than a specific type of model; it is separate from attention and can be applied to all types of neural models, including attention-based models.

Our work is closely related to the idea of adversarial example generation (Szegedy et al., 2013; Nguyen et al., 2015). Szegedy et al. (2013) train a model to fool the network by generating adversarial images. Similarities and differences between the proposed approach and adversarial example generation will be detailed in Section 4.2.

# 3 Linking Word Vector Dimensions to Linguistic Features

It is well accepted that vector representation dimensions encode human-interpretable features, such as part-of-speech tags and syntactic features, and can help many tagging tasks (Collobert et al., 2011). However, it is unclear how the information of these human-developed features is encoded in vector dimensions and, more importantly, how a tagging model extracts the information.

To answer the questions above, we examine various feature classification models to analyze how a neural model extracts information from word vector dimensions and why it makes specific tagging decisions. We conduct experiments on features that are widely adopted in NLP tasks, including part of speech (POS), named entity class (NER), chunking, prefix, suffix, word-shape and word-frequency. We first train models on benchmarks with gold-standard labels for these features. Then we rationalize a model's decision by analyzing the effect of representation erasure (both input word vectors and intermediate hidden units).

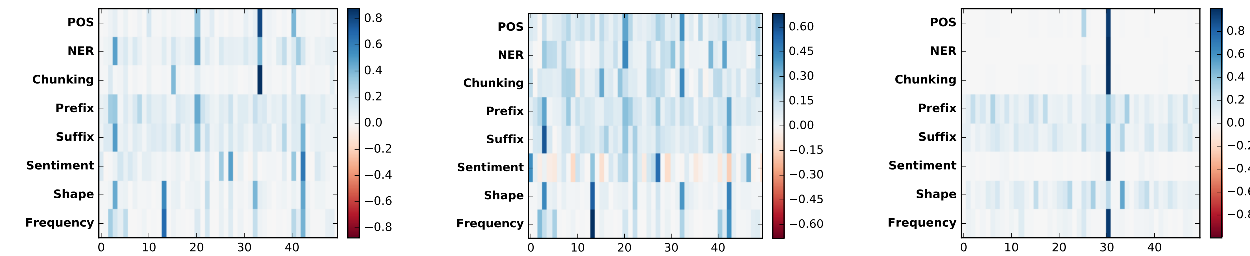
## 3.1 Visualization Model

Let $M$ denote a neural model (already trained). Given a training example $e \in E$ with gold-standard label $c$, with $L_e$ denoting the index of the tag for $e$, the log-likelihood assigned by model $M$ to the correct label for $e$ is denoted by $S(e, c) = -\log P(L_e = c)$. Now let $d$ be the index of some vector dimension we are interested in exploring, and let $S(e, c, \neg d)$ denote the log-likelihood of the correct label for $e$ according to $M$ if dimension $d$ is erased; that is, its value set to 0. The *importance* of dimension $d$—denoted by $I(d)$—is the relative difference between $S(e, c)$ and $S(e, c, \neg d)$:

$$I(d) = \frac{1}{|E|} \sum_{e \in E} \frac{S(e, c) - S(e, c, \neg d)}{S(e, c)} \qquad (1)$$

## 3.2 Tasks and Training

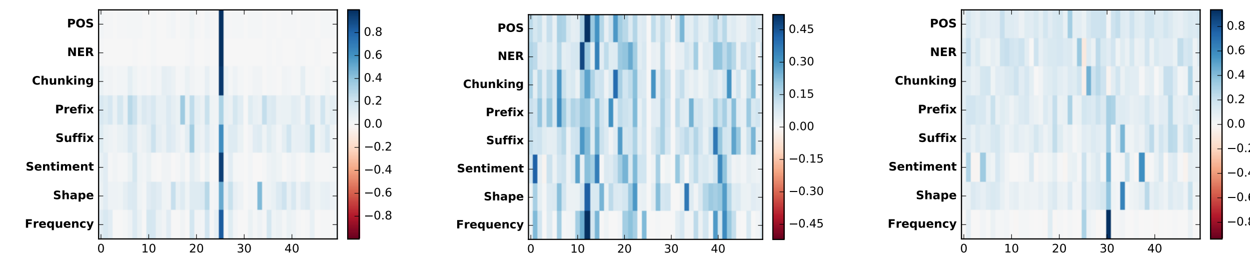
The tasks we consider are divided into two categories: (1) Sequence tagging tasks, in which the model needs to assign labels to all tokens within a sentence. In sequence tagging tasks, the tag of a word depends on the word itself and its neighbors. The sequence tagging tasks we consider include POS tagging, NER




(a) Word2vec, no dropout.

(b) Word2vec, with dropout.

(c) GloVe, no dropout.




(d) GloVe, no dropout; 31rd dimension removed.

(e) GloVe, no dropout; 31rd, 26th dimensions removed.

(f) GloVe, with dropout.

Figure 1: Heatmap of word vector dimension importance $I(d)$, computed using Eq. 1, for different training strategies and word vectors. Each cell shows the importance of a dimension (column) on each task (row) for the trained model. Accuracy numbers for each training strategy are shown in Table 5 in the Appendix.

tagging and chunking. (2) Word ontology classification tasks, in which no context needs to be considered. The input is just the word itself, and the model needs to assign a label to the input word. Tasks we consider in this category include prefix, suffix, sentiment and word-shape classification as well as (log) word-frequency prediction.

The details of each task are presented in Table 6 in the Appendix. For sequence tagging tasks, the input consists of the concatenation of the vector representation of the word to tag and the representations of its neighbors (window size is set to 5). For ontology tagging tasks, the input is just the representation of the input word. We focus on two types of word vectors, word2vec (Mikolov et al., 2013b) and GloVe (Pennington et al., 2014). We use 50-dimensional vectors pre-trained using the Gigaword-Wiki corpus. For each task, we train a four-layer neural model (an input word-embedding layer, 2 intermediate layers, and a output layer that outputs a scalar) using a structure similar to that of Collobert et al. (2011) with a TANH activation function. Each intermediate layer contains 50 hidden units. Test accuracy for each task is shown in Table 5 in the Appendix.

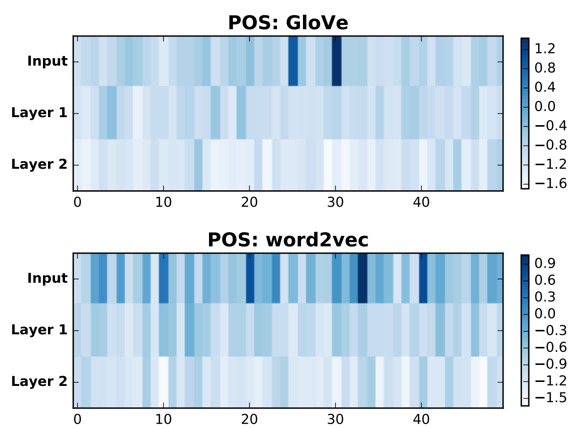



Figure 2: Heatmap of importance (computed using Eq. 1) of each layer for the POS task. Each cell corresponds a unit in a neural model layer. Each column denotes a dimension and each row denotes a layer in the network. Importance values are projected to log space.

### 3.3 Results

For each task, we take the pre-trained model, erase an input word dimension by setting its value to 0, apply the pre-trained model to the modified inputs, and apply Eq. 1 to compute the importance score of the erased dimension.

Results are shown in Figure 1. Each row corre-

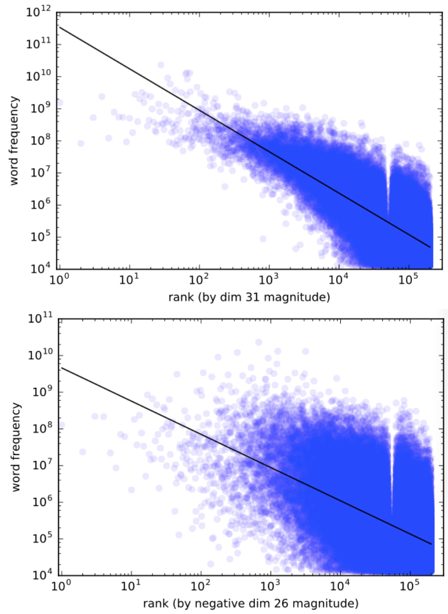



Figure 3: Correlation with word frequency of the magnitude of (a) the 31st dimension ($R^2 = 0.55$, $p < 1 \times 10^{-5}$) and (b) the 26th dimension ($R^2 = 0.27$, $p < 1 \times 10^{-5}$) of GloVe vectors.

sponds a feature classification task (e.g., POS, NER) and each column in a row signifies the importance of a word-vector dimension to the pre-trained model for that task. For word2vec vectors (shown in Figure 1a), we observe clear patterns that the model focuses more on some dimensions than others and that some tasks share important dimensions. For example, POS and chunking share dimension 34; NER, prefix and suffix share dimensions 4 and 31; etc. When applying dropout (Srivastava et al., 2014), we can clearly see that importance is distributed more equally among different dimensions, which is intuitive since the model is forced to make use of other dimensions when the dominating dimension is dropped during training.

Things are a bit more confusing with Glove vectors (Figure 1c): we observe a single dimension (d31) dominating across almost all tasks. Interestingly, if we remove dimension d31 and retrain the model, another dominant dimension (d26) appears (Figure 1d). Only if we remove both these dimensions (Figure 1e) can the model spread its attention to most of the other dimensions. Interestingly, performance does not drop after removing these two dimensions and retraining the models (as shown in Table 5 in the Appendix).

In Figure 1f, which shows the effects of using dropout, these two dimensions (26 and 31) still stand out in the frequency regression task despite the fact that their influence declines dramatically in other tasks. Such a phenomenon indicates that these two dimensions very likely correlate with word frequency. We thus rank words by dimension magnitude and calculate the correlation between word frequency and dimension values in Figure 3. We can see a significant correlation between the values of the 26th and 31st dimensions and word frequency. Our explanation for the behavior of the model trained on GloVe vectors is as follows: since word frequency is an important feature for many classification tasks, the trained model tends to attach strong focus to the frequency-based dimensions. However, dimensions other than the word-frequency dimensions also bear enough information for the model to make correct decisions. This is why when the word-frequency dimensions are eliminated, the model still performs as competitively. It is also worth noting that word2vec vectors don't bear the same strong word-frequency information as GloVe vectors. We conjecture that this is because tokens are omitted in proportion to word-type frequency in word2vec models (Mikolov et al., 2013a).

Figure 2 shows importance values for hidden unit dimensions in different layers on the POS task. We present results for other tasks in Figure 6 in the Appendix. The heatmap color is generally lighter in the higher layers, which means importance is distributed more equally among different dimensions on higher layers. The more equal distribution of importance among higher layer dimensions makes the importance scores generally smaller in higher layers than lower ones, which means the final decision is more robust to the change of one particular dimension in higher layers. Such a phenomenon fits with our expectation that a neural model tends to distill information from a few important dimensions in the input layer, making the removal of these input layer dimensions more detrimental. At higher layers, however, the information is spread across different units.

## 4 Words and Phrases in Sentiment Analysis

The section above is concerned mostly with individual vector dimensions. However, for most tasks in

| rank | Bi-LSTM | Uni-LSTM | RNN |
|---|---|---|---|
| 1 | masterpiece (104) | masterpiece (32.0) | pathetic (8.25) |
| 2 | sweetest (46.9) | dreadful (31.7) | dreadful (6.23) |
| 3 | dreadful (43.8) | sweetest (13.9) | brilliant (5.63) |
| 4 | stillborn (20.7) | pathetic (9.75) | ungainly (4.58) |
| 5 | pathetic (16.5) | flawless (7.77) | smartest (4.41) |
| 6 | eye-popping (13.1) | breathtaking (6.66) | hated (4.33) |
| 7 | succeeds (12.8) | dumbness (6.57) | eye-popping (4.06) |
| 8 | breathtaking (12.4) | beaut (6.25) | stupider (3.40) |
| 9 | ugliest (9.81) | disappointingly (6.20) | dicey (3.30) |
| 10 | flawless (9.64) | heady (6.08) | masterpiece (3.30) |

Table 1: Top 10 ranked words by importance (computed using Eq. 1) from the Bi-LSTM, Uni-LSTM and standard RNN models.

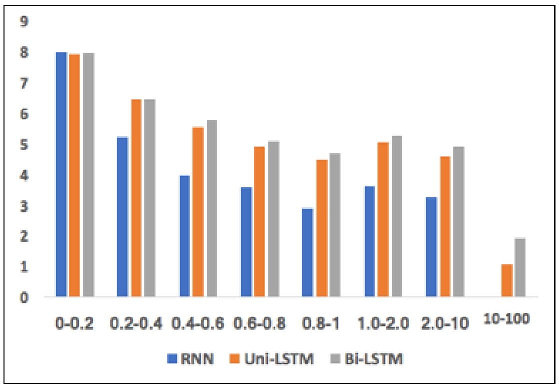



Figure 4: Histogram of words by importance for different models.

NLP, words rather than individual dimensions function as basic units. In this section, we demonstrate how the proposed model can facilitate the understanding of neural models at the word level. We focus on sentiment classification tasks and consider two datasets: the Stanford Treebank dataset (Socher et al., 2013), which focuses on phrase/sentence level classification, and a hotel review dataset crawled from TripAdvisor for aspect rating classification on longer documents.

### 4.1 Individual Word Importance for Sentiment

We can compute the importance of words similarly to that of word-vector dimensions, by calculating the relative change of the log-likelihood of the correct sentiment label for a text unit when a particular word is erased. The formula is exactly the same as Eq. 1, but with dimensions replaced by words.

We examine three models: a *standard RNN* with TANH activation functions, an LSTM (*Uni-LSTM*) and a bidirectional LSTM (*Bi-LSTM*), all trained on the Stanford treebank dataset. We first transform each parse tree constituent in the dataset to a sequence of tokens. Each sequence is then mapped to a phrase/sentence representation and fed to a softmax classifier. The *Bi-LSTM*, *Uni-LSTM* and standard *RNN* respectively obtain an accuracy of 0.526, 0.501 and 0.453 on sentence-level fine-grained classification. It is worth noting that the *Bi-LSTM* model achieves state-of-the-art performance in sentence-level fine-grained classification on this benchmark, significantly outperforming tree-based models, namely 50.1 reported in Zhu et al. (2015) and 51.0 in Tai et al. (2015)[1]. We refer the readers to the Appendix for more details about the dataset and model training.

We present the importance scores of a few selected sentiment-indicative words in Table 3. The ranking score is computed by averaging the log-likelihood difference resulting from erasing that word across all test examples containing the word. We can see that the *Bi-LSTM* is more sensitive to the deletion of these sentiment indicators than the *Uni-LSTM*, which is in turn more sensitive than the *RNN*. This is presumably due to the gate structures in LSTMs that control information flow, making these architectures better at focusing on words that indicate sentiment.

The highest-ranked words by importance (computed using Eq.1) for each model are listed in Table 1 (more comprehensive lists are presented in Table 8 in the Appendix). Figure 4 shows a histogram of all words by importance for different models. The distribution also confirms that the *Bi-LSTM* model is more sensitive to the sentiment-indicative words, with more words in buckets with higher importance values.

Figure 5 plots the importance score of individual

[1]Tai et al. (2015) reported an accuracy of 49.0 for Bi-LSTM models, significantly underperforming our implementations.

| rank | Word | Score | Label | Original Sentence |
|---|---|---|---|---|
| 1 | revelatory | -0.90 | - | flat, but with a revelatory performance by michelle williams. |
| 2 | lacks | -0.88 | + | what it lacks in originality it makes up for in intelligence and b-grade stylishness. |
| 3 | shame | -0.84 | + | it takes this never-ending confusion and hatred, puts a human face on it, evokes shame among all who are party to it and even promotes understanding. |
| 4 | skip | -0.83 | + | skip work to see it at the first opportunity. |
| 5 | lackadaisical | -0.82 | + | a pleasant ramble through the sort of idoosyncratic terrain that errol morris has often dealt with... it does possess a loose, lackadaisical charm. |
| 6 | by-the-books | -0.82 | + | a fairly by-the-books blend of action and romance with sprinklings of intentional and unintentional comedy. |
| 7 | misses | -0.82 | ++ | this is cool, slick stuff, ready to quench the thirst of an audience that misses the summer blockbusters. |
| 8 | bonehead | -0.82 | ++ | the smartest bonehead comedy of the summer. |
| 9 | dingy | -0.81 | + | it's a nicely detailed world of pawns, bishops and kings, of wagers in dingy backrooms or pristine forests. |
| 10 | enjoying | -0.81 | - | i kept thinking over and over again,' i should be enjoying this.' |
| 11 | foul | -0.80 | + | a whole lot foul, freaky and funny. |
| 12 | best | -0.80 | - | the best way to hope for any chance of enjoying this film is by lowering your expectations. |
| 25 | pleasing | -0.72 | - | an intermittently pleasing but mostly routine effort. |

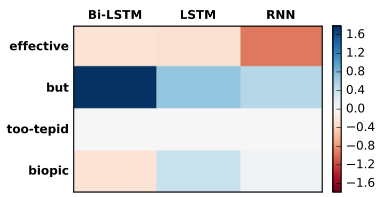
Table 2: Words with high negative importance score (computed using Eq.1) obtained by the Bi-LSTM model. Negative importance means that the model makes better prediction when the word is erased. ++, +, 0, -, – respectively denote strong positive, positive, neutral, negative and strong negative sentiment labels. which are gold-standard ones from the dataset.

| word | Bi-LSTMs | Uni-LSTMs | RNN |
|---|---|---|---|
| greatest | 9.463 | 5.593 | 0.742 |
| wonderful | 9.521 | 3.292 | 0.704 |
| worst | 7.739 | 4.698 | 0.967 |
| excellent | 6.835 | 4.883 | 1.859 |
| best | 4.916 | 2.448 | 0.548 |
| hated | 6.557 | 3.512 | 4.338 |
| love | 1.678 | 1.786 | 0.999 |
| unforgettable | 2.286 | 1.648 | 1.482 |
| waste | 4.579 | 3.600 | 2.342 |
| disaster | 3.728 | 3.362 | 0.021 |

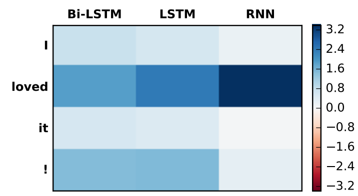
Table 3: Importance score (computed using Eq.1) for a few sentiment indicators assigned by different models.

words (rows) for the different models (columns) in a few specific examples of sentence-level sentiment classification. Higher values mean that the model is more sensitive to the erasing of a particular word. As can be seen, all three models attach more importance to words that are indicative of sentiment (e.g., "loved", "entertainment", "greatest") and dampen the influence of other tokens. LSTM-based models generally show a clearer focus on sentiment words than standard RNN models, and they also succeed in attaching importance to intensification tokens (e.g., the exclamation mark in Figure 5b), which the *RNN* fails to identify.

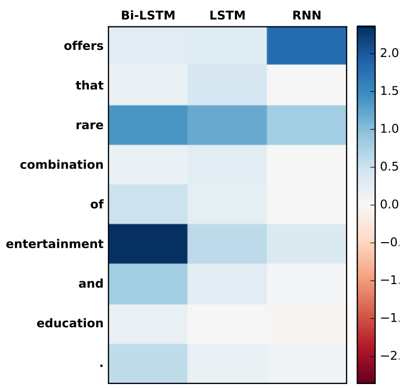
We also notice an interesting phenomenon in Figure 5: the importance scores of words can take negative values, which means that the removal of some words actually improves the model's decision. Such discoveries can help with error analysis on a model by identifying which words confuse the model and lead to mistakes. We therefore also list the top-ranked words by negative importance score (the removal of which words can best help the model make the correct decision). We present some of the top negative important words obtained using the Bi-LSTM model in Table 2, while listing comprehensive results from all three models in Tables 9, 10 and 11 in the Appendix. From these tables, we can clearly identify a few patterns that make neural models fail: (1) A common sentiment indicator word is used in a context (e.g., describing details of the movie) that makes the word not bear any sentiment orientation, such as the word *happy* in *happy ending* (Figure 5e), or *shame* (Table 2, rank 3). (2) A sentiment indicator word is used in a specific context that turns its sentiment into the opposite of its common usage; e.g., "the smartest bonehead" (Table 2, rank 8). (3) A sentiment indicator is used in the scope of a model verb—e.g., *i should be enjoying this* (Table 2.rank10)—or in an ironic context—e.g., *the best way to hope for any chance of enjoying this film is by lowering your expectations* (Table 2, rank 12). (4) A sentiment indicator is used in a concessive sentence, requiring the handling of discourse information; e.g., *revelatory* in *flat, but with a revelatory performance by michelle williams* (Table 2, rank 1), *pleasing* in *an intermit-*



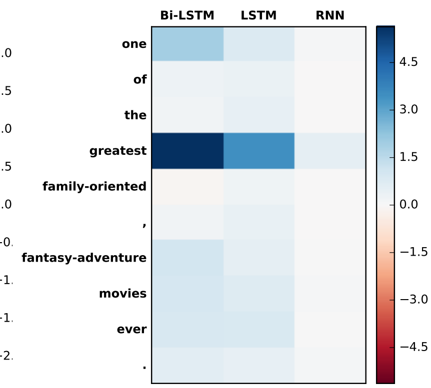

(a) Neutral



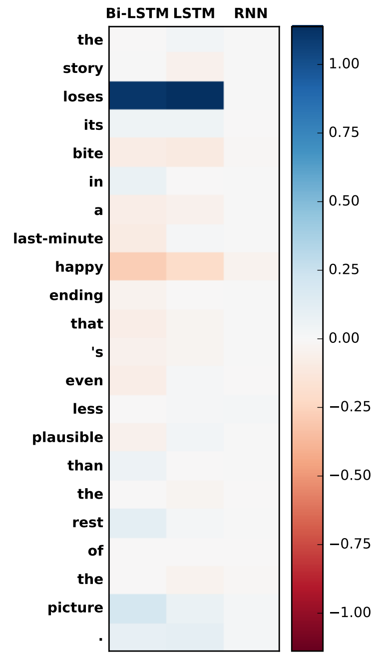

(b) Strong positive




(c) Strong positive




(d) Strong positive




(e) Strong negative

Figure 5: Heatmap of word importance (computed using Eq. 1) in sentiment analysis.

*tently pleasing but mostly routine effort* (Table 2, rank 25). Resolving these problems is a long-term goal of future work in sentiment analysis.

### 4.2 Looking at multiple words

The analysis that we have described so far deals with individual words or dimensions, by removing one word at one time to see its importance. However, in many NLP tasks, words do not function independently. They compose to form the meaning of large text units (e.g., phrases, sentences). Looking at each word individually might lead to losing sight of a global picture of how a neural model makes decisions.

Still under the methodology of representation erasure, we propose another strategy to interpret neural network decisions: removing the minimum number of words to change the model's prediction[2] . The problem can be defined more formally as follows: let $e$ denote an input text unit consisting of a sequence of words, $e = \{w_1, w_2, ..., w_N\}$, where $N$ denotes the number of words in $e$, and let $L_e$ denote index of the label that $M$ gives to $e$. The task is to discover a minimal subset of $e$, denoted by $D \subset e$, such that the removal of all words in $D$ from $e$ (the remaining words are denoted by $e - D$) will change the label $L_e$. Let $|D|$ denote the number of words in D. The problem is formalized as follows

$$\min_{D} |D| \quad s.t. \;\; L_{e-D} \neq L_e \tag{2}$$

Finding the optimal solution requires enumerating all different word combinations, which is computationally intractable when the number of words in $e$ gets large. To address this issue, we propose an strategy based on reinforcement learning to find an approximate solution.

Given a pre-trained sentiment classification model

[2]The proposed model is closely related to the idea of adversarial example generation (Szegedy et al., 2013; Nguyen et al., 2015), which is to find the minimal change to input dimensions to change the neural network decision, but is different in that (1) adversarial training is usually not suited for interpreting how a model makes decision but for detecting the intrinsic flaws of neural models. Adversarial examples are usually very similar to real examples (sometimes adversarial images cannot even recognized by human in Computer Vision) but can fool the model to make a different decision. (2) Words operate as a basic unit in most tasks of NLP. Changing individual dimensions may break the integrity of text units (e.g., break the language model). The proposed model attempts to remove words rather than make changes to individual dimensions, which makes the proposed problem discrete as opposed to being continuous in adversarial example generation.

$M$, an input example $e$, and the label $L_e$ that $M$ gives to $e$, we define a policy $\pi$ over a binary variable $z_t$, indicating whether a word $w_t \in e$ should be removed. $z_t$ takes the value of 1 when $w_t$ is removed and 0 otherwise. The policy model takes as input the representation associated with word $w$ at the current time step outputted from model $M$ and defines a binary distribution $\pi$ over $z_w$. The policy model examines every word in $e$ and decides whether the word should be kept or removed ($D$ denotes the union of the removed words). After the policy model finishes removing words from $e$, the pre-trained sentiment model $M$ gives another label $L_{e-D}$ to the remaining words $e - D$.

To train the policy model, a reward function is necessary. The policy model receives a reward of 1 if the label is changed, i.e., $L^*_{e-D} \neq L_e$, and 0 if the label remains the same. Since we not only want the label to be changed, but also want to find the minimal set of words to change the label, the reward is scaled by the number of the words that are removed. This means removing more words will be rewarded less than removing fewer words if both of them the change the classification label. We therefore propose the following reward:

$$L(e, D) = \frac{1}{|D|} \cdot \mathbf{1}(L_{e-D} \neq L_e) \tag{3}$$

We also add a regularizer that encourages similar values of $z$ for words within the same sentence to encourage (or discourage) leaving out contiguous phrases:

$$\Omega(e, z) = \gamma \sum_{s \in S} \sum_{t \in s} |z_t - z_{t-1}| \tag{4}$$

where $S$ denotes the collection of sentences by breaking the input $e$. Such an idea is inspired by group lasso (Meier et al., 2008), which has been widely employed in many NLP tasks, such as document classification (Yogatama and Smith, 2014) and providing rationales for neural model interpretation (Lei et al., 2016). Let $R(e) = L(e, D) - \Omega(z_{1:N})$ denote the final reward. The system is trained to maximize the expected reward of the sequence of erasing/not-erasing decisions:

$$J(\theta) = \mathbb{E}_\pi(R(e)|\theta) \tag{5}$$

The gradient of (5) is approximated using the likelihood ratio trick (Williams, 1992; Glynn, 1990; 0), in which for a given $e$, we sample a sequence of decisions based on $\pi$, compute the associated reward and backward propagate gradients to update $\pi$, which can be summarized as follows:

$$\nabla J(\theta) \approx \nabla \log \cdot \pi(z_{1:N}|\theta)(R(e) - b(e)) \tag{6}$$

Here $b(e)$ denotes a baseline value, to reduce the variance of the estimate while keeping it unbiased.[3]

The policy model is trained to interpret the pre-trained sentiment classification model. Therefore, during the RL training, the original sentiment model is kept fixed.

**Task, Dataset and Training** Inspired by recent visualization work from Lei et al. (2016), we focus on the task of document-level aspect rating prediction (Tang et al., 2015a; Tang et al., 2015b). We collected hotel reviews from TripAdvisor. The dataset contains roughly 870,000 reviews with an average length of 120 words. Each review contains ranking scores (integers from 1 to 5) for different aspects of the hotel, such as service, cleanliness, location, rooms, etc. We choose the aspect sentiment classification task because each review might contain diverse sentiments towards different aspects, and it is interesting to see how a model manages or fails to identify these different aspects and their associated scores when entangled with other aspects. We focus on four aspects: value, rooms, service and location.

Since the sentiment correlation between any pair of aspects (and the overall score) is high, the result of which may confuse the model, we employ a strategy similar to that of Lei et al. (2016) to pick less correlated examples. For a given aspect, we pick the 50,000 reviews for which the score of this aspect deviates the most from the mean of the other aspects. We use two different models to map input reviews to vector representations: a vanilla Bi-LSTM and a memory-network structure (Sukhbaatar et al., 2015) similar to Tang et al. (2016) with attention at both

[3] To estimate the baseline value, we train another neural network model to estimate the reward of input $e$ under current policy $\pi$, similar to Ranzato et al. (2015). The network takes as input $e$, transforms it to a vector representation using an LSTM and maps the representation to a scalar. The network is optimized based on the mean squared loss between the estimated reward and real reward. The policy $\pi$ and the baseline $t$ can be viewed as actor and critic in an actor-critic architecture (Sutton and Barto, 1998).

| |
|---|
| (1) clean updated room. friendly efficient staff . rate was too high 199 plus they charged 10 day for internet access in the room . |
| (2) the location is fantastic. the staff are helpful and service oriented . sleeping rooms meeting rooms and public lavatories not cleaned on a daily basis . the hotel seems a bit old and a bit tired overall . trolley noise outside can go into the wee hours . if you get a great price for a few nights this hotel may be a good choice . breakfast is very nice remember if you just stick to the cold buffet it is cheaper . |
| (3) location is nice . but goes from bad to worse once you walk through the door . staff very surly and unhelpful . room and hallway had a very strange smell . rooms very run down . so bad that i checked out immediately and went to another hotel . intercontinental chain should be ashamed . |
| (4) i took my daughter and her step sister to see a show at webster hall . it is so overpriced i 'm in awe . i felt safe . the rooms were tiny . lots of street noise all night from the partiers at the ale house below . |

(a) Examples of minimal set of erased words based on *Bi-LSTM* model

| |
|---|
| (1) clean updated room. friendly efficient staff . rate was too high 199 plus they charged 10 day for internet access in the room . |
| the location is fantastic. the staff are helpful and service oriented . (2) sleeping rooms meeting rooms and public lavatories not cleaned on a daily basis . the hotel seems a bit old and a bit tired overall . trolley noise outside can go into the wee hours . if you get a great price for a few nights this hotel may be a good choice . breakfast is very nice remember if you just stick to the cold buffet it is cheaper . |
| (3) location is nice . but goes from bad to worse once you walk through the door . staff very surly and unhelpful . room and hallway had a very strange smell . rooms very run down . so bad that i checked out immediately and went to another hotel . intercontinental chain should be ashamed . |
| (4) i took my daughter and her step sister to see a show at webster hall . it is so overpriced i 'm in awe . i felt safe . the rooms were tiny . lots of street noise all night from the partiers at the ale house below . |

(b) Examples of minimal set of erased words based on *memory-network* model.

Table 4: Examples of minimal set of erased words to change the model decision for different aspects based on different models. Each of the colors represents a specific aspect, i.e., rooms, service, value and location.

word level and sentence level. Model accuracies are shown in Table 7 in Appendix.

The representation is then fed to a 5-class softmax function. Given a trained $M$, we then train (with RL) a policy to discover the minimal set of words to erase to flip the model's classification decision.

### 4.3 Results

Sampled results are presented in Table 4. We can see that the reinforcement learning model identifies important phrases that indicate the sentiment of different aspects, providing a rationale for why the sentiment model makes a certain decision. By comparing Table 4a with Table 4b, we can see that the reinforcement model trained based on the memory-based model offers better interpretability than the one trained based on *LSTMs*. The latter model not only requires erasing more words to flip the model's decision, but also sometimes deletes passages describing different aspects or overall sentiment. Since the RL model is trained based on the representations outputted from the sentiment model, better interpretability of the RL model indicates the superiority of the sentiment model.

## 5 Conclusion

In this paper, we propose a general methodology for interpreting neural network decisions by analyzing the effect of erasing particular representations to see how such a behavior affects a neural model's decision. We observe that the proposed framework offers many interpretable explanations for various aspects of neural models and offers a way to conduct error analysis on neural model decisions.

## References

V. M. Aleksandrov, V. I. Sysoyev, and V. V. Shemeneva. 1968. Stochastic optimization. *Engineering Cybernetics*, 5:11–16.

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

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(1):3.

Lingjia Deng and Janyce Wiebe. 2015. Mpqa 3.0: An entity/event-level sentiment corpus. In *Conference of* *the North American Chapter of the Association of Computational Linguistics: Human Language Technologies*.

Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2014. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

Manaal Faruqui, Yulia Tsvetkov, Dani Yogatama, Chris Dyer, and Noah Smith. 2015. Sparse overcomplete word vector representations. *arXiv preprint arXiv:1506.02004*.

Alona Fyshe, Leila Wehbe, Partha P Talukdar, Brian Murphy, and Tom M Mitchell. 2015. A compositional and interpretable semantic space. *Proceedings of the NAACL-HLT, Denver, USA*.

Peter W Glynn. 1990. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84.

Klaus Greff, Rupesh Kumar Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2015. Lstm: A search space odyssey. *arXiv preprint arXiv:1503.04069*.

Aurélie Herbelot and Eva Maria Vecchi. 2015. Building a shared world: Mapping distributional to model-theoretic semantic spaces. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 22–32.

Michiel Hermans and Benjamin Schrauwen. 2013. Training and analysing deep recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 190–198.

Thorsten Joachims. 2002. *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

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

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. *arXiv preprint arXiv:1606.04155*.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2015. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066*.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Lukas Meier, Sara Van De Geer, and Peter Bühlmann. 2008. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Brian Murphy, Partha Pratim Talukdar, and Tom Mitchell. 2012. Learning effective and interpretable semantic models using non-negative sparse embedding. Association for Computational Linguistics.

Anh Nguyen, Jason Yosinski, and Jeff Clune. 2015. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 427–436. IEEE.

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 (EMNLP)*, pages 1532–1543.

MarcAurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*.

Xing Shi, Kevin Knight, and Deniz Yuret. 2016. Why neural translations are the right length. *EMNLP*.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642. Citeseer.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Hendrik Strobelt, Sebastian Gehrmann, Bernd Huber, Hanspeter Pfister, and Alexander M Rush. 2016. Visual analysis of hidden state dynamics in recurrent neural networks. *arXiv preprint arXiv:1606.07461*.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Richard S Sutton and Andrew G Barto. 1998. *Introduction to reinforcement learning*, volume 135. MIT Press Cambridge.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob

Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Duyu Tang, Bing Qin, and Ting Liu. 2015a. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432.

Duyu Tang, Bing Qin, Ting Liu, and Yuekui Yang. 2015b. User modeling with neural network for review rating prediction. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 1340–1346. AAAI Press.

Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. *arXiv preprint arXiv:1605.08900*.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Huijuan Xu and Kate Saenko. 2015. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. *arXiv preprint arXiv:1511.05234*.

Dani Yogatama and Noah A Smith. 2014. Linguistic structured sparsity in text categorization.

Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. 2015. Long short-term memory over recursive structures. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1604–1612.

# 6 Appendix

## 6.1 Dataset Statistics and Training Accuracy for Feature Classification in Section 3

- POS Tagging: Each word is associated with a unique tag that indicates its syntactic role, such as plural noun, adverb, etc. We follow the standard Penn Treebank split, using sections 0-18/19-21/22-24 as training/dev/test sets, respectively.

- NER Tagging: Each word is associated with a named entity tag, such as "person" or "location". We evaluate on the CoNLL-2003 shared benchmark dataset for NER (Tjong Kim Sang and De Meulder, 2003).

- Chunking: Each word is assigned only one unique tag, encoded as a begin-chunk (e.g. B-NP) or inside-chunk tag (e.g. I-NP). We use the CoNLL-2000 dataset, in which sections 15-18 of WSJ data are used for training and section 20 for testing. Validation is performed by splitting the training set.

- Prefix and Suffix: Words are segmented using the Morfessor package (Creutz and Lagus, 2007). We retained top 250 frequent prefixes and suffixes. Other than "s" and numbers, single characters are abandoned. We kept a list of 200,000 most frequent words, 51,083 of which are matched with a prefix and 78,804 of which are matched with a suffix. We split words into train/dev/test splits in the ratio 0.8/0.1/0.1.

- Sentiment: We use the MPQA subjectivity lexicon list (Deng and Wiebe, 2015; Wilson et al., 2005), which consists roughly 8,000 lexicons.

- Word shape: words are mapped to X, XX, XXX, etc. based on the number characters it contains.

- Word frequency: the number of word occurrences is computed using a Wikipedia dump and is then mapped to log space. Unlike all the others, which are multi-class classification tasks, word-frequency prediction is a regression task: minimize the mean squared error predicting the log frequency of each word.

A summary of the datasets is given in Table 6.

Test accuracy/error for different training strategies presented in Figure 1 are shown in Table 5. For classification tasks (i.e., POS, NER, Chunking, Prefix, Suffix, Sentiment, Word Shape), we report accuracy; higher values of accuracy are better. For the regression task (Frequency), we report the Mean Squared Loss (loss for short); lower values of loss are better.

| Training Strategy | Vector | POS | NER | Chunk | Prefix | Suffix | Sentiment | Shape | Freq |
|---|---|---|---|---|---|---|---|---|---|
| Vanilla (Figure 1c) | GloVe | 0.912 | 0.954 | 0.921 | 0.334 | 0.208 | 0.857 | 0.256 | **0.349** |
| d31 removed (Figure 1d) | GloVe | **0.915** | 0.954 | 0.921 | 0.336 | 0.207 | 0.818 | **0.259** | 0.355 |
| d31, d26 removed (Figure 1e) | GloVe | 0.914 | **0.959** | **0.923** | **0.339** | 0.209 | **0.860** | 0.250 | 0.413 |
| Dropout 0.2 (Figure 1f) | GloVe | 0.857 | 0.953 | 0.907 | 0.317 | **0.239** | 0.820 | 0.240 | 0.861 |
| Vanilla (Figure 1a) | word2vec | 0.911 | 0.954 | 0.918 | 0.301 | 0.161 | 0.826 | 0.236 | 1.059 |
| Dropout 0.2 (Figure 1a) | word2vec | 0.889 | 0.952 | 0.893 | 0.289 | 0.154 | 0.819 | 0.224 | 1.486 |

Table 5: Testing accuracy for different training strategies on tagging tasks.

| Task | #Training | #Dev | #Test | #Class |
|---|---|---|---|---|
| POS | 875,462 | 126,419 | 124,202 | 45 |
| NER | 189,403 | 47,959 | 42,723 | 6 |
| Chunk | 203,359 | 20,336 | 45,470 | 3 |
| Prefix | 41,406 | 4,601 | 5,076 | 250 |
| Suffix | 63,946 | 7,106 | 7,752 | 250 |
| Sentiment | 4,950 | 551 | 446 | 3 |
| Shape | 89,864 | 8,987 | 10,126 | 22 |
| Frequency | 123,235 | 13,693 | 15,050 | – |

Table 6: Statistics of datasets for dimension visualization tasks.

### 6.2 Stanford Sentiment Treebank and Training Detail

The Stanford Sentiment Treebank is a benchmark dataset widely used for neural model evaluations. The dataset contains gold-standard sentiment labels for every parse tree constituent, from sentences to phrases to individual words, for a total of 215,154 phrases in 11,855 sentences. The task is to perform both fine-grained (very positive, positive, neutral, negative and very negative) and coarse-grained (positive vs. negative) classification at both the phrase and sentence level.

### 6.3 Aspect Rating Prediction

The results for aspect rating prediction using the two models along with other baselines are shown in Table 7. Feature based SVM models are trained using SVM-light package (Joachims, 2002). LSTM based models do not perform as competitively as simple bigram-based classification models in aspect classification tasks, which has also been observed in Tang et al. (2016).

| Aspect | service | location | rooms | rooms |
|---|---|---|---|---|
| SVM+Unigram | 40.1 | 53.8 | 42.0 | 39.0 |
| SVM+Bigram | 43.2 | 53.1 | 41.1 | 46.1 |
| Bi-LSTM | 37.5 | 51.4 | 29.8 | 30.5 |
| Tang et al., (2016) | 43.2 | 54.0 | 39.4 | 38.0 |

Table 7: Results for aspect rating classification (5-class) from different models.

| rank | bi-lstm | uni-lstm | rnn |
|---|---|---|---|
| 11 | wonderful (9.529) | stillborn (5.939) | flawless (3.250) |
| 12 | bastard (9.464) | pleasurably (5.857) | heart-stopping (3.231) |
| 13 | greatest (9.460) | savor (5.854) | unwatchable (3.204) |
| 14 | brilliant (8.350) | succeeds (5.767) | tremendous (2.832) |
| 15 | worst (7.739) | punch-drunk (5.728) | lukewarm (2.820) |
| 16 | excellent (6.835) | inert (5.594) | cop-out (2.684) |
| 17 | dumbness (6.633) | greatest (5.593) | drab (2.654) |
| 18 | nicely (6.605) | irresistible (5.461) | incredible (2.419) |
| 19 | hated (6.557) | nicely (5.444) | sweetest (2.365) |
| 20 | lukewarm (6.443) | brilliant (5.361) | waste (2.342) |
| 21 | unpleasant (6.355) | skillfully (5.158) | overstuffed (2.320) |
| 22 | clunker (5.754) | must-see (5.125) | vulgar (2.312) |
| 23 | cop-out (5.712) | excellent (4.883) | lackluster (2.269) |
| 24 | beaut (5.650) | bothersome (4.728) | bothersome (2.256) |
| 25 | beautiful (5.401) | worst (4.698) | punish (2.119) |
| 26 | must-see (5.362) | heart-stopping (4.347) | pleasurably (2.085) |
| 27 | deliciously (5.226) | refreshing (4.225) | muted (1.863) |
| 28 | sabotages (5.103) | invigorating (4.137) | excellent (1.859) |
| 29 | irresistible (4.977) | travesty (3.977) | fabulous (1.853) |
| 30 | best (4.916) | fabulous (3.949) | dumbness (1.834) |
| 31 | incredible (4.914) | eye-popping (3.919) | stupidest (1.777) |
| 32 | stupider (4.810) | incredible (3.875) | flaccid (1.768) |
| 33 | fabulous (4.585) | imaginative (3.823) | clunker (1.756) |
| 34 | waste (4.579) | deliciously (3.802) | ridiculous (1.717) |
| 35 | disappointingly (4.565) | misses (3.722) | suffocated (1.653) |
| 36 | tremendous (4.302) | incarnates (3.694) | sorry (1.624) |
| 37 | bothersome (4.212) | waste (3.600) | jackasses (1.579) |
| 38 | pet (4.141) | hated (3.512) | turgid (1.544) |
| 39 | misses (4.139) | feast (3.442) | snoozer (1.515) |
| 40 | wannabe (4.072) | snoozer (3.368) | unforgettable (1.482) |
| 41 | repulsive (4.071) | disaster (3.362) | deliciously (1.418) |
| 42 | bracing (4.043) | wonderful (3.298) | blandness (1.399) |
| 43 | ingenious (4.019) | stupider (3.243) | delightful (1.337) |
| 44 | moot (3.981) | clunker (3.231) | exasperating (1.320) |
| 45 | invigorating (3.940) | mesmerizing (3.185) | failed (1.317) |
| 46 | snoozer (3.928) | lukewarm (3.163) | roller-coaster (1.287) |
| 47 | punch-drunk (3.877) | rent (3.120) | suffer (1.264) |
| 48 | overstuffed (3.831) | worthwhile (3.034) | achievement (1.224) |
| 49 | unwatchable (3.785) | superior (2.964) | nowhere (1.175) |
| 50 | delight (3.766) | letdown (2.959) | remarkable (1.168) |
| 51 | breezes (3.747) | hollow (2.925) | breathtaking (1.162) |
| 52 | joyful (3.734) | screenwriters (2.862) | beaut (1.161) |
| 53 | disaster (3.728) | ugliest (2.816) | awfully (1.135) |
| 54 | ungainly (3.710) | moot (2.804) | sour (1.125) |
| 55 | pleasurably (3.710) | astute (2.767) | hilarious (1.124) |
| 56 | exquisitely (3.633) | thoughtful (2.757) | monotonous (1.110) |
| 57 | marvellous (3.587) | vile (2.734) | inviting (1.104) |
| 58 | hilarious (3.533) | repulsive (2.721) | treat (1.085) |
| 59 | travesty (3.500) | likeable (2.693) | worthwhile (1.066) |
| 60 | sparkles (3.469) | bravely (2.691) | mesmerizing (1.055) |

Table 8: Top ranked words by importance (computed using Eq. 1) from the Uni-LSTM, Bi-LSTM and standard RNN models.

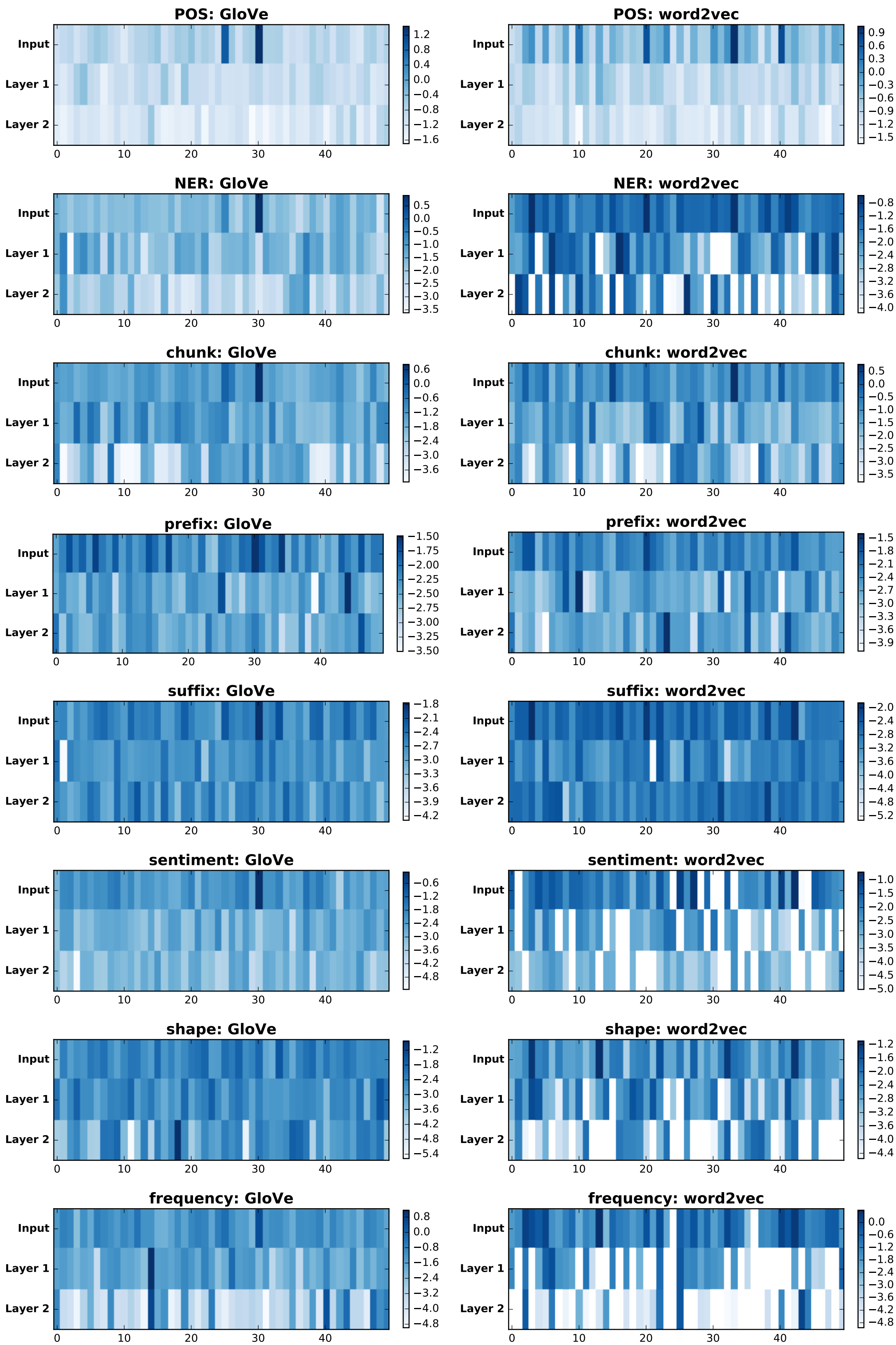



Figure 6: Heatmap of importance (computed using Eq. 1) of each layer for different tasks. Each column denotes a dimension and each row denotes a layer in the network. Importance values are transformed to log space.

| rank | Word | Score | Label | Original Sentence |
|---|---|---|---|---|
| 1 | revelatory | -0.90 | - | flat, but with a revelatory performance by michelle williams. |
| 2 | lacks | -0.88 | + | what it lacks in originality it makes up for in intelligence and b-grade stylishness. |
| 3 | shame | -0.84 | + | it takes this never-ending confusion and hatred, puts a human face on it, evokes shame among all who are party to it and even promotes understanding. |
| 4 | skip | -0.83 | + | skip work to see it at the first opportunity. |
| 5 | lackadaisical | -0.82 | + | a pleasant ramble through the sort of idoosyncratic terrain that errol morris has often dealt with... it does possess a loose, lackadaisical charm. |
| 6 | by-the-books | -0.82 | + | a fairly by-the-books blend of action and romance with sprinklings of intentional and unintentional comedy. |
| 7 | misses | -0.82 | ++ | this is cool, slick stuff, ready to quench the thirst of an audience that misses the summer blockbusters. |
| 8 | bonehead | -0.82 | ++ | the smartest bonehead comedy of the summer. |
| 9 | dingy | -0.81 | + | it's a nicely detailed world of pawns, bishops and kings, of wagers in dingy backrooms or pristine forests. |
| 10 | enjoying | -0.81 | - | i kept thinking over and over again,' i should be enjoying this.' |
| 11 | best | -0.80 | - | the best way to hope for any chance of enjoying this film is by lowering your expectations. |
| 12 | skillful | -0.79 | + | a slick, skillful little horror film. |
| 13 | confident | -0.79 | - | just when the movie seems confident enough to handle subtlety, it dives into soapy bathos. |
| 14 | inconsequential | -0.77 | + | has a shambling charm... a cheerfully inconsequential diversion. |
| 15 | oblivion | -0.77 | ++ | ... mesmerizing, an eye-opening tour of modern beijing culture in a journey of rebellion, retreat into oblivion and return. |
| 16 | captivating | -0.76 | + | a captivating cross-cultural comedy of manners. |
| 17 | acidic | -0.75 | ++ | hilarious, acidic brit comedy. |
| 18 | overblown | -0.75 | ++ | occasionally funny, always very colorful and enjoyably overblown in the traditional almodvar style. |
| 19 | n't | -0.75 | ++ | a sensitive and expertly acted crowd-pleaser that is n't above a little broad comedy and a few unabashedly sentimental tears. |
| 20 | n't | -0.75 | ++ | a great comedy filmmaker knows great comedy need n't always make us laugh. |
| 21 | inconsequential | -0.75 | + | has a shambling charm... a cheerfully inconsequential diversion. |
| 22 | entertaining | -0.74 | + | sturdy, entertaining period drama... both caine and fraser have their moments. |
| 23 | south-of-the-border | -0.74 | - | like a south-of-the-border melrose place. |
| 24 | not | -0.74 | ++ | it's a good film – not a classic, but odd, entertaining and authentic. |
| 25 | pleasing | -0.72 | - | an intermittently pleasing but mostly routine effort. |
| 26 | difficult | -0.72 | ++ | a worthy entry into a very difficult genre. |
| 27 | lost | -0.72 | + | gets under the skin of a man who has just lost his wife. |
| 28 | dahmer | -0.72 | ++ | renner's performance as dahmer is unforgettable, deeply absorbing. |
| 29 | by-the-numbers | -0.72 | + | " antwone fisher" is an earnest, by-the-numbers effort by washington. |
| 30 | great | -0.72 | - | it's a great deal of sizzle and very little steak. |
| 31 | exuberantly | -0.72 | + | zany, exuberantly irreverent animated space adventure. |
| 32 | dumb | -0.71 | + | the transporter is as lively and as fun as it is unapologetically dumb |
| 33 | fascinating | -0.71 | + | it's both a necessary political work and a fascinating documentary... |
| 34 | insultingly | -0.71 | + | it is so refreshing to see robin williams turn 180 degrees from the string of insultingly innocuous and sappy fiascoes he's been making for the last several years. |
| 35 | none | -0.71 | + | as a witness to several greek-american weddings – but, happily, a victim of none – i can testify to the comparative accuracy of ms. vardalos' memories and insights. |
| 36 | squalor | -0.71 | + | the result is mesmerizing – filled with menace and squalor. |
| 37 | insurance | -0.70 | - | technically, the film is about as interesting as an insurance commercial. |
| 38 | mess | -0.70 | - | just a bloody mess. |
| 39 | inviting | -0.70 | - | we are left with a superficial snapshot that, however engaging, is insufficiently enlightening and inviting. |
| 40 | comedy | -0.70 | + | a pleasant romantic comedy. |
| 41 | bravura | -0.70 | - | a bravura exercise in emptiness. |
| 42 | n't | -0.70 | 0 | in the end, white oleander is n't an adaptation of a novel. |
| 43 | well | -0.70 | - | well, it does go on forever. |
| 44 | unhappily | -0.70 | + | happily for mr. chin – though unhappily for his subjects – the invisible hand of the marketplace wrote a script that no human screenwriter could have hoped to match. |
| 45 | slugs | -0.70 | 0 | melanie eventually slugs the yankee. |
| 46 | good | -0.69 | - | an ultra-low-budget indie debut that smacks more of good intentions than talent. |
| 47 | time-wasting | -0.69 | 0 | ... an agreeable time-wasting device – but george pal's low-tech 1960 version still rules the epochs. |
| 48 | departure | -0.69 | + | not for everyone, but for those with whom it will connect, it's a nice departure from standard moviegoing fare. |

Table 9: Words ranked by negative importance score (computed using Eq. 1) for the Bi-LSTM model. Negative importance means that the model makes better predictions when the word is erased. ++, +, 0, -, and -- respectively denote strong positive, positive, neutral, negative and strong negative gold-standard labels from the dataset.

| rank | Word | Score | Label | Original Sentence |
|---|---|---|---|---|
| 1 | foul | -0.89 | + | a whole lot foul, freaky and funny. |
| 2 | shaky | -0.84 | + | as shaky as the plot is, kaufman's script is still memorable for some great one-liners. |
| 3 | bonehead | -0.83 | ++ | the smartest bonehead comedy of the summer. |
| 4 | harsh | -0.81 | + | harsh, effective documentary on life in the israeli-occupied palestinian territories. |
| 5 | lacks | -0.80 | + | what it lacks in originality it makes up for in intelligence and b-grade stylishness. |
| 6 | skip | -0.79 | + | skip work to see it at the first opportunity. |
| 7 | confident | -0.79 | - | just when the movie seems confident enough to handle subtlety, it dives into soapy bathos. |
| 8 | shame | -0.78 | + | it takes this never-ending confusion and hatred, puts a human face on it, evokes shame among all who are party to it and even promotes understanding. |
| 9 | claptrap | -0.78 | + | more a load of enjoyable, conan-esque claptrap than the punishing, special-effects soul assaults the mummy pictures represent. |
| 10 | wonderful | -0.76 | - | while benigni -lrb- who stars and co-wrote -rrb- seems to be having a wonderful time, he might be alone in that. |
| 11 | dingy | -0.74 | + | it's a nicely detailed world of pawns, bishops and kings, of wagers in dingy backrooms or pristine forests. |
| 12 | great | -0.74 | - | it's a great deal of sizzle and very little steak. |
| 13 | bogus | -0.73 | + | -lrb- a -rrb- hollywood sheen bedevils the film from the very beginning... -lrb- but -rrb- lohman's moist, deeply emotional eyes shine through this bogus veneer... |
| 14 | engrossing | -0.73 | - | where last time jokes flowed out of cho's life story, which provided an engrossing dramatic through line, here the comedian hides behind obviously constructed routines. |
| 15 | preposterous | -0.72 | + | while the isle is both preposterous and thoroughly misogynistic, its vistas are incredibly beautiful to look at. |
| 16 | stunning | -0.71 | + | hayek is stunning as frida and... a star-making project. |
| 17 | camouflaged | -0.71 | + | a film of precious increments artfully camouflaged as everyday activities. |
| 18 | dumb | -0.70 | + | the transporter is as lively and as fun as it is unapologetically dumb |
| 19 | dahmer | -0.70 | ++ | renner's performance as dahmer is unforgettable, deeply absorbing. |
| 20 | disturbing | -0.70 | ++ | disturbing and brilliant documentary. |
| 21 | marvelous | -0.70 | + | marvelous, merry and, yes, melancholy film. |
| 22 | enjoyable | -0.70 | + | an enjoyable film for the family, amusing and cute for both adults and kids. |
| 23 | thankfully | -0.69 | + | farrell... thankfully manages to outshine the role and successfully plays the foil to willis's world-weary colonel. |
| 24 | pleasing | -0.69 | - | an intermittently pleasing but mostly routine effort. |
| 25 | half-wit | -0.69 | ++ | an enjoyably half-wit remake of the venerable italian comedy big deal on madonna street. |
| 26 | brimful | -0.69 | 0 | brimful. |
| 27 | time-wasting | -0.69 | 0 | ... an agreeable time-wasting device – but george pal's low-tech 1960 version still rules the epochs. |
| 28 | nothing | -0.68 | + | sometimes, nothing satisfies like old-fashioned swashbuckling. |
| 29 | by-the-books | -0.68 | + | a fairly by-the-books blend of action and romance with sprinklings of intentional and unintentional comedy. |
| 30 | misses | -0.68 | ++ | this is cool, slick stuff, ready to quench the thirst of an audience that misses the summer blockbusters. |
| 31 | soggy | -0.67 | - | a soggy, cliche-bound epic-horror yarn that ends up being even dumber than its title. |
| 32 | worse | -0.67 | 0 | every so often a movie comes along that confirms one's worse fears about civilization as we know it. |
| 33 | no | -0.67 | 0 | no question. |
| 34 | captivating | -0.67 | + | a captivating cross-cultural comedy of manners. |
| 35 | worse | -0.66 | 0 | it's a worse sign when you begin to envy her condition. |
| 36 | shambling | -0.66 | + | has a shambling charm... a cheerfully inconsequential diversion. |
| 37 | modest | -0.65 | - | a modest and messy metaphysical thriller offering more questions than answers. |
| 38 | captivating | -0.65 | + | most of crush is a clever and captivating romantic comedy with a welcome pinch of tartness. |
| 39 | intelligent | -0.65 | + | a mostly intelligent, engrossing and psychologically resonant suspenser. |
| 40 | sterile | -0.65 | + | a distant, even sterile, yet compulsively watchable look at the sordid life of hogan's heroes star bob crane. |
| 41 | terrific | -0.65 | 0 | the actors are so terrific at conveying their young angst, we do indeed feel for them. |
| 42 | unconcerned | -0.64 | + | here's a british flick gleefully unconcerned with plausibility, yet just as determined to entertain you. |
| 43 | intriguing | -0.64 | - | kwan makes the mix-and - match metaphors intriguing, while lulling us into torpor with his cultivated allergy to action. |
| 44 | slugs | -0.64 | 0 | melanie eventually slugs the yankee. |
| 45 | south-of-the-border | -0.63 | - | like a south-of-the-border melrose place. |
| 46 | terrific | -0.63 | + | highlights are the terrific performances by christopher plummer, as the prime villain, and nathan lane as vincent crummles, the eccentric theater company manager. |
| 47 | mournfully | -0.63 | + | noyce's film is contemplative and mournfully reflective. |

Table 10: Words ranked by negative importance score (computed using Eq. 1) obtained by the Uni-LSTM model. Negative importance means that the model makes better prediction when the word is erased. ++, +, 0, -, and -- respectively denote strong positive, positive, neutral, negative and strong negative gold-standard sentiment labels from the dataset.

| rank | Word | Score | Label | Original Sentence |
|---|---|---|---|---|
| 1 | effective | -0.93 | 0 | effective but too-tepid biopic |
| 2 | no | -0.89 | 0 | no question. |
| 3 | high | -0.87 | 0 | high on melodrama. |
| 4 | brimful | -0.86 | 0 | brimful. |
| 5 | bravura | -0.85 | - | a bravura exercise in emptiness. |
| 6 | pleasing | -0.84 | - | an intermittently pleasing but mostly routine effort. |
| 7 | stunning | -0.83 | + | hayek is stunning as frida and... a star-making project. |
| 8 | n't | -0.80 | + | is n't it great ? |
| 9 | engaging | -0.80 | + | an engaging overview of johnson's eccentric career. |
| 10 | thrill | -0.80 | - | the thrill is -lrb- long -rrb- gone. |
| 11 | real | -0.79 | -- | a real clunker. |
| 12 | captivating | -0.79 | + | a captivating cross-cultural comedy of manners. |
| 13 | skip | -0.79 | + | skip work to see it at the first opportunity. |
| 14 | insomnia | -0.78 | 0 | insomnia is involving. |
| 15 | right | -0.78 | 0 | oh, it's extreme, all right. |
| 16 | faultlessly | -0.78 | 0 | faultlessly professional but finally slight. |
| 17 | well | -0.78 | 0 | well before it's over, beijing bicycle begins spinning its wheels. |
| 18 | fun | -0.77 | 0 | as a director, mr. ratliff wisely rejects the temptation to make fun of his subjects. |
| 19 | well | -0.77 | - | well, it does go on forever. |
| 20 | absorbing | -0.77 | + | an absorbing, slice-of-depression life that touches nerves and rings true. |
| 21 | no | -0.77 | + | finally, a genre movie that delivers – in a couple of genres, no less. |
| 22 | good | -0.77 | - | first good, then bothersome. |
| 23 | harsh | -0.76 | + | harsh, effective documentary on life in the israeli-occupied palestinian territories. |
| 24 | great | -0.76 | - | it's a great deal of sizzle and very little steak. |
| 25 | good | -0.75 | + | bullock does a good job here of working against her natural likability. |
| 26 | no | -0.74 | + | by the end of no such thing the audience, like beatrice, has a watchful affection for the monster. |
| 27 | terrific | -0.73 | 0 | the actors are so terrific at conveying their young angst, we do indeed feel for them. |
| 28 | best | -0.72 | - | my response to the film is best described as lukewarm. |
| 29 | newton | -0.71 | ++ | newton draws our attention like a magnet, and acts circles around her better known co-star, mark wahlberg. |
| 30 | best | -0.71 | - | the best way to hope for any chance of enjoying this film is by lowering your expectations. |
| 31 | community | -0.71 | -- | it feels like a community theater production of a great broadway play : even at its best, it will never hold a candle to the original. |
| 32 | hmm | -0.71 | 0 | hmm. |
| 33 | invincible | -0.71 | + | the invincible werner herzog is alive and well and living in la |
| 34 | departure | -0.71 | ++ | greene delivers a typically solid performance in a role that is a bit of a departure from the noble characters he has played in the past, and he is matched by schweig, who carries the film on his broad, handsome shoulders. |
| 35 | self-aware | -0.71 | - | it's fairly self-aware in its dumbness. |
| 36 | no | -0.70 | ++ | waydowntown is by no means a perfect film, but its boasts a huge charm factor and smacks of originality. |
| 37 | happy | -0.70 | 0 | just what makes us happy, anyway ? |
| 38 | vivid | -0.70 | - | the essential problem in orange county is that, having created an unusually vivid set of characters worthy of its strong cast, the film flounders when it comes to giving them something to do. |
| 39 | understands | -0.70 | 0 | ... understands that a generation defines its music as much as the music defines a generation. |
| 40 | no | -0.70 | + | this version's no classic like its predecessor, but its pleasures are still plentiful. |
| 41 | pleasant | -0.69 | + | a pleasant romantic comedy. |
| 42 | community | -0.69 | -- | it feels like a community theater production of a great broadway play : even at its best, it will never hold a candle to the original. |
| 43 | gimmick | -0.69 | ++ | an endearingly offbeat romantic comedy with a great meet-cute gimmick. |
| 44 | community | -0.69 | -- | it feels like a community theater production of a great broadway play : even at its best, it will never hold a candle to the original. |
| 45 | love | -0.69 | 0 | hip-hop has a history, and it's a metaphor for this love story. |
| 46 | great | -0.69 | -- | if melville is creatively a great whale, this film is canned tuna. |
| 47 | clever | -0.69 | + | a clever blend of fact and fiction. |
| 48 | demeanor | -0.69 | -- | the smug, oily demeanor that donovan adopts throughout the stupidly named pipe dream is just repulsive. |
| 49 | decent | -0.68 | - | some decent actors inflict big damage upon their reputations. |
| 50 | slugs | -0.68 | 0 | melanie eventually slugs the yankee. |

Table 11: Words ranked by negative importance score (computed using Eq.1) obtained by the RNN model. Negative importance means that the model makes better prediction when the word is erased. ++, +, 0, -, and -- respectively denote strong positive, positive, neutral, negative and strong negative gold-standard sentiment labels from the dataset.