

# Segmentation and Classification of Opinions with Recurrent Neural Networks

Imran Sheikh, Irina Illina, Dominique Fohr

► **To cite this version:**

Imran Sheikh, Irina Illina, Dominique Fohr. Segmentation and Classification of Opinions with Recurrent Neural Networks. IEEE Information Systems and Economic Intelligence, May 2017, Al Hoceima, Morocco. hal-01491182

**HAL Id: hal-01491182**

**<https://hal.inria.fr/hal-01491182>**

Submitted on 16 Mar 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Segmentation and Classification of Opinions with Recurrent Neural Networks

Imran Sheikh, Irina Illina, Dominique Fohr

Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France

Inria, Villers-lès-Nancy, F-54600, France

CNRS, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France

{imran.sheikh, irina.illina, dominique.fohr}@loria.fr

**Abstract**—Automatic opinion/sentiment analysis is essential for analysing large amounts of text as well as audio/video data communicated by users. This analysis provides highly valuable information to companies, government and other entities, who want to understand the likes, dislikes and feedback of the users and people in general. Opinion/Sentiment analysis can follow a classification approach or perform a detailed aspect level analysis. In this paper, we address a problem in between these two, that of segmentation and classification of opinions in text. We propose a recurrent neural network model with bi-directional LSTM-RNN, to perform joint segmentation and classification of opinions. We introduce a novel method to train neural networks for segmentation tasks. With experiments on a dataset built from the standard RT movie review dataset, we demonstrate the effectiveness of our proposed model. Proposed model gives promising results on opinion segmentation, and can be extended to general sequence segmentation tasks.

## I. INTRODUCTION

With the growing amount of users on the internet, social media and online shopping websites, a large amount of data is generated in which people voluntarily publish their opinion on products, stocks, policies, etc. Automatic systems are necessary to analyse such large data and derive facts from them. Accordingly, the area of automatic opinion/sentiment analysis is receiving interest from both industry and academia, with some challenges and tasks being held every year [1]–[4]. Research in sentiment analysis involves building systems and algorithms which can understand text from the perspective of the opinions or sentiments expressed in it [5], [6]. Sentiment analysis systems are very useful for industries to obtain a feedback on their products which get reviewed on social networks and online shopping websites [7]–[10]. Similarly they have been used for analysing sentiments in political tweets and election data [11]–[14]. Apart from text data, videos posted on the social media and news websites [15]–[19], as well as audio conversations from call centres [20]–[22], are analysed for sentiments.

A common task is to classify a given sentence or text as expressing positive or negative sentiments using a text categorization approach [23]. On the other hand, classifying sentiments at the sentence and document level may not provide a detailed analysis of the entity or product being reviewed. For example, the movie review - *The actors did their job but the characters are simply awesome*, attributes different level of opinions to different entities. In such cases aspect level

sentiment analysis [24] is performed. Aspect level sentiment analysis is concerned with identification of sentiment-target pairs in the text, their classification and also the aggregation over each aspect to provide a concise overview. (Although in practice, a method may not implement all these steps and in the same order.)

We focus on a sentiment analysis problem which is in-between sentence/document level classification and aspect level sentiment analysis. We investigate the problem of analysis of text which can contain segments corresponding to both positive and negative sentiments, with one following the other. For example, the review - *Comes with a stunning camera and high screen resolution. Quick wireless charging but the battery life is a spoiler*. Given such a text, our task is to segment (and classify) the text into parts of positive and negative segments. So, the goal is to automatically identify the segment '*Comes with a stunning camera and high screen resolution. Quick wireless charging*', classify it as positive, and have the segment '*but the battery life is a spoiler*' classified with negative sentiments. Similar to our previous example, in aspect level sentiment analysis, the text can have different sentiment-target pairs or aspects. However, our task is segmentation and classification of text based on opinions/sentiments, without performing a detailed aspect level analysis.

An important point to note is that, the segmentation models cannot simply rely on sentence boundaries, punctuations or any other linguistic features. As in the example we presented, the segment boundaries are not always at the end of the sentence. Another common scenario where the segmentation models cannot rely on such features is sentiment analysis on audio/video data. *Automatic Speech Recognition* (ASR) transcripts of these audio/video documents are used for sentiment analysis and they do not contain any kind of punctuation marks. To perform a robust segmentation and classification under such conditions, we propose discriminatively trained neural network models.

Developments in neural networks and deep learning has led to new state-of-the-art results in text and language processing tasks. Text classification is being commonly performed with compositional representations learned with neural networks or by training the network specifically for text classification [25]. Fully connected feed forward neural networks [26]–[28], *Convolutional Neural Networks* (CNN) [29]–[31] and also

*Recurrent/Recursive Neural Networks* (RNN) [32]–[36] have been used successfully. The approaches based on CNN and RNN capture rich compositional information and have been outperforming previous results on standard tasks in natural language processing. Of particular interest to this work are the Recurrent Neural Networks (RNN). Previous works have shown that RNNs are very good at modelling word sequences in text. Since our task is to segment a text sequence and classify the segments into positive and negative sentiments, we exploit RNNs to perform the segmentation and classification of opinions.

In this paper, we present our approach to train RNNs for segmentation and classification of opinions in text. We propose a novel cost function to train the RNN in a discriminative manner to perform the segmentation (Equation (12)). We evaluate our proposed model on the task of segmentation and classification of movie reviews into positive and negative sentiments. The dataset used in our experiments is built using sentences from the standard *Rotten Tomatoes* (RT) movie review dataset [37]. The rest of the paper is organised as follows. In Section II we present the previous works related to our work. We provide a description of RNNs in Section III. Our proposed model is presented in Section IV. Section V describes our experiment setup, including the experiment dataset, model training and evaluation measures. This is followed by a discussion on the results obtained with our model in Section V-C and the conclusion in Section VI.

## II. RELATED WORK

Classical sentiment analysis approaches traditionally relied on classifiers trained using sentiment specific lexicons or knowledge structures and other hand crafted features [5], [6]. These classical feature based approaches also tried neural networks for sentiment classification [38]. Maas et. al. [39] proposed models which automatically learned word features (or word vector representations) targeted for the sentiment classification. Neural network models and automatically learned word vector features came together to achieve state-of-the-art results on sentiment classification with the model proposed in [26]. Later, the application of deep learning techniques in natural language processing led to new state-of-the-art results on sentiment classification tasks [40], mainly with CNN [29], [30] and RNN [36] architectures. Neural network architectures, including both CNN and RNN, have been also applied for aspect level sentiment analysis [2], [41], [42].

In this paper, we explore RNN models for segmentation and classification of opinions/sentiments in text. As mentioned earlier, our task is in-between sentiment classification and aspect level sentiment analysis. Compared to earlier works using RNN for sentiment analysis [36], [43]–[45], we propose a novel method for discriminative training of RNNs for joint text segmentation and classification. Text segmentation approaches have been studied vastly in previous works. These approaches scan the text and determine the locations of segment cuts/boundaries based on coherence calculated between adjacent blocks of text. After the initial work in this area

[46], [47], most approaches used topic models and Bayesian approaches for text segmentation task [48]–[50]. We employ word embeddings [51], vector representations of words which carry both syntactic and semantic information of words and the context, to learn sentiment level cohesion of segments in text. As opposed to the methods relying on Bayesian and topic models, our approach can detect very short segments containing only few words.

More recently word embeddings from neural network models have been utilised for text segmentation in general [52] as well as specifically for sentiment analysis [53]. The approach in [52] uses an iterative refinement technique whereas the work in [53] is focused on finding appropriate phrase like segments which imply the correct sentiments (for example *not good* actually implies bad). In contrast to these approaches, we exploit *Long Short-Term Memory* (LSTM) RNNs to capture and remember the sentiment level cohesion, mainly to perform appropriate segmentation and classification of opinions and sentiments in text.

## III. RECURRENT NEURAL NETWORKS

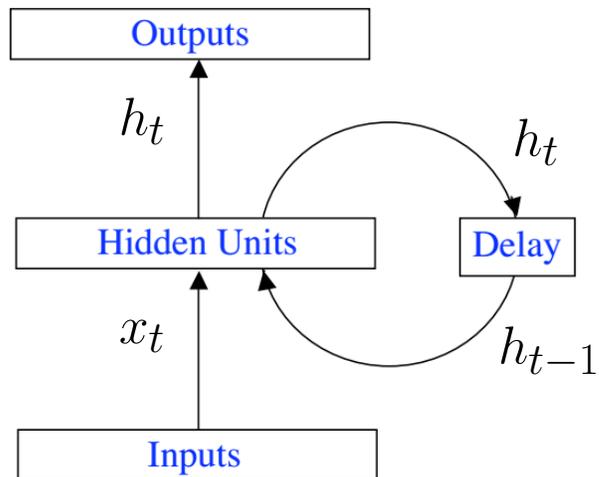


Fig. 1. Schema of Recurrent Neural Network (RNN)

Figure 1 shows a schema for a typical Recurrent Neural Network (RNN) (the most commonly used Elman network [54]). Similar to most neural networks, RNN has an input layer, a layer of hidden units and an output layer. Given a discrete input sequence  $\{x_t\}_{t=1,2,3,\dots,N}$ , a hidden layer activation  $h_t$ , is obtained using the current input  $x_t$  and the previous hidden layer activation  $h_{t-1}$ . The corresponding output  $y_t$  is then obtained using  $h_t$ . The computation of the hidden layer and output activations is given as:

$$h_t = f_x(x_t.W_x + h_{t-1}.W_h + b_x) \quad (1)$$

$$y_t = f_y(h_t.W_y + b_y) \quad (2)$$

where  $W_x$ ,  $W_h$  and  $W_y$  are the weight parameters at input, hidden and output layers,  $b_x$  and  $b_y$  are the bias parameters

at input and output layers,  $f_x$  and  $f_y$  denote non-linearity functions like sigmoid and hyperbolic tangent ( $\tanh$ ). Training the RNN involves learning the weight and bias parameters. Given a training dataset with input sequences and output labels, this can be achieved using gradient descent and error back propagation algorithms [55], [56].

As it is evident from Figure 1, the RNN can remember activations of the past inputs. This enables it to model sequences like discrete time sequences in speech signals, word sequences in a document, protein sequences, etc. However, training RNN requires error back propagation through time and as the length of the sequence/time increase it leads to vanishing and exploding gradient problems [57]. To address the more severe vanishing gradient problem, the Long Short-Term Memory (LSTM) cell [58] has become a popular alternative to the hidden layer unit in the classical RNN.

Figure 2 shows an illustration of the LSTM cell. The LSTM cell borrows ideas from a hardware memory cell, and as shown in the figure it consists of a cell state  $c_t$  and a forget gate  $f_t$  which controls the amount of past activations to be memorised and/or to be forgotten by the cell. The computations of the activations at the input gate ( $i_t$ ), forget gate  $f_t$ , cell state ( $c_t$ ), output gate ( $o_t$ ) and the hidden layer ( $h_t$ ) are given as:

$$i_t = \sigma(x_t.W_i + h_{t-1}.U_i + b_i) \quad (3)$$

$$\tilde{c}_t = \tanh(x_t.W_c + h_{t-1}.U_c + b_c) \quad (4)$$

$$f_t = \sigma(x_t.W_f + h_{t-1}.U_f + b_f) \quad (5)$$

$$c_t = i_t * \tilde{c}_t + f_t * c_{t-1} \quad (6)$$

$$o_t = \sigma(x_t.W_o + h_{t-1}.U_o + c_t.V_o + b_o) \quad (7)$$

$$h_t = o_t * \tanh(c_t) \quad (8)$$

where  $W$ ,  $U$  and  $b$  are weight parameters and bias with suffixes  $i$ ,  $f$ ,  $c$  and  $o$  denoting input gate, forget gate, cell state and output gate, respectively.  $\sigma$ ,  $\tanh$  denote the sigmoid and hyperbolic tangent non-linearities and  $V_o$  is another weight parameter at the output gate.  $*$  denotes simple (element-wise) multiplication and  $\cdot$  denotes vector/matrix dot products in a multi-dimensional setup.

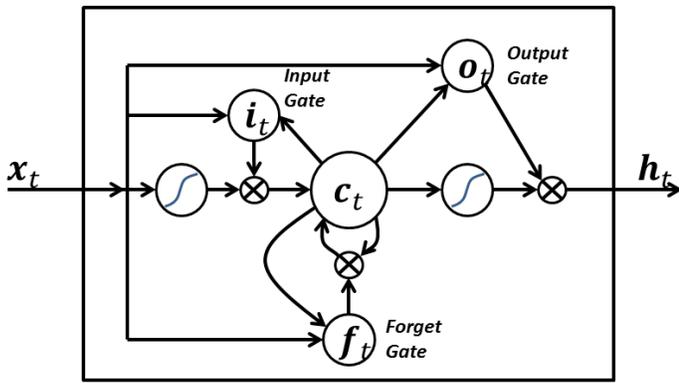


Fig. 2. Long Short-Term Memory (LSTM) Cell. (Taken from [59].)

#### IV. PROPOSED APPROACH

We propose a model based on LSTM-RNN to perform joint segmentation and classification of opinions/sentiments in text. More specifically we employ a bidirectional LSTM-RNN, in which there is a forward LSTM-RNN which models the word sequence from left to right and a backward LSTM-RNN which models the word sequence from right to left. In previous works, bidirectional LSTM-RNN are shown to perform better than unidirectional LSTM-RNN for modelling and classifying sequences [60]–[62]. However, we specifically choose bidirectional LSTM-RNN to compare and measure cohesion between past and future word sequences. In our task, it will enable us to detect changes in sentiments as well as in context, and hence to perform segmentation. At the same time, the activations from the bidirectional LSTM-RNN will be combined to perform a classification of sentiments corresponding to the segments.

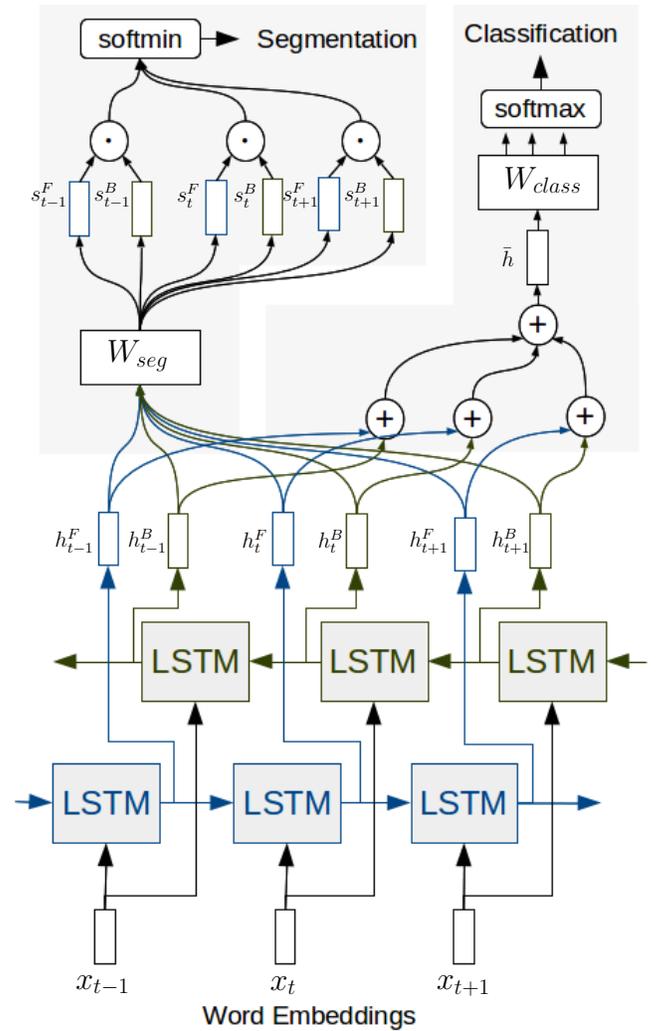


Fig. 3. Proposed model for joint Segmentation and Classification

Figure 3 shows the diagrammatic representation of our proposed model for joint segmentation and classification. As

shown in the figure, the model operates on word embeddings corresponding to words in a text sequence (denoted as  $\dots, x_{t-1}, x_t, x_{t+1}, \dots$ ). Following a bidirectional LSTM-RNN architecture, our model has two layers of LSTM-RNN. The hidden layer activation for the forward LSTM-RNN at time  $t$  is denoted as  $h_t^F$  and the hidden layer activation for the backward LSTM-RNN at time  $t$  is denoted as  $h_t^B$ . Compared to Figure 1, here the RNN schema is unrolled across  $t$ . The hidden layer activations of the forward and backward LSTM-RNN are obtained using Equations (3)-(8). These hidden layer activations ( $h_t^F, h_t^B$ ) are used in both the segmentation and classification sub-parts, shown in top part of the Figure 3.

We first present the segmentation part of our model, which is shown in the top left part of Figure 3. Each of the hidden layer activations of the forward and backward LSTM-RNN are transformed using a feed forward neural network as follows:

$$s_t^F = h_t^F \cdot W_{seg} + b_{seg} \quad (9)$$

$$s_t^B = h_t^B \cdot W_{seg} + b_{seg} \quad (10)$$

where  $W_{seg}$  and  $b_{seg}$  are weight and bias parameters of the segmentation feed forward network. Following the feed forward layer, the outputs corresponding to forward and backward LSTM-RNN at each  $t$  are compared as:

$$d_t = s_t^F \cdot s_t^B \quad (11)$$

where  $\cdot$  denotes a dot product. This dot product compares the similarity between the context until  $t$ , as captured in  $s_t^F$ , and the context following  $t$ , which is captured in  $s_t^B$ . Thus  $\{d_t\}_{t=1:N}$ , where  $N$  is the length of the text sequence, represents similarity across the text sequence and this similarity should be minimum at the segment boundaries. The similarity calculation is followed by a softmax function, given as:

$$\text{softmax}(d_{t*}) = \frac{e^{-d_{t*}}}{\sum_{t=1}^N e^{-d_t}} \quad (12)$$

The softmax function will give highest output probability to the lowest  $d_t$ . Additionally it will enable a discriminative training of the segmentation model, by maximising the likelihood of the true segmentation point ( $t^*$ , known at time of training), as compared to all the other points ( $t = 1, 2, \dots, N$ ).

Given the output of segmentation, the opinions/sentiments in the text segments can be classified using separate models which are trained for sentiment classification [27], [29], [30], [36], [63]. However, we would like to study the power of our model for joint segmentation and classification. Thus, in addition to the segmentation part, our model also has a classification part as shown in top right of Figure 3. The hidden layer activations of the forward and backward LSTM-RNN are averaged to form a single vector representation  $\bar{h}$  of the entire text, as:

$$\bar{h} = \frac{1}{2N} \left( \sum_{t=1}^N h_t^F + \sum_{t=1}^N h_t^B \right) \quad (13)$$

This vector representation of the text is then fed into a feed forward neural network as follows:

$$\hat{y}_t = \bar{h} \cdot W_{class} + b_{class} \quad (14)$$

TABLE I  
EXAMPLE OF REVIEWS FROM THE RT DATASET

Review	Sentiment Polarity
<i>an exhilarating experience.</i>	positive
<i>over-the-top and a bit ostentatious, this is a movie that's got oodles of style and substance.</i>	positive
<i>decent but dull.</i>	negative
<i>i suspect that there are more interesting ways of dealing with the subject.</i>	negative

where  $W_{class}$  and  $b_{class}$  are the weight and bias parameters of the classification feed forward network. This is followed by a softmax function, given as:

$$\text{softmax}(\hat{y}_{l*}) = \frac{e^{\hat{y}_{l*}}}{\sum_{l=1}^L e^{\hat{y}_l}} \quad (15)$$

The softmax function will give highest output probability to the highest  $\hat{y}_t$ . Additionally it will enable a discriminative training of the classification model, by maximising the likelihood of the true sentiment/opinion classes ( $l^*$ , known at time of training), as compared to all possible classes. In our experiments, we perform segmentation and classification of text containing two opinionated segments, which can be categorised into four classes as -, +, +-, ++ (with - denoting negative and + denoting positive).

## V. EXPERIMENTS AND RESULTS

In this section we first present a description of the dataset used in our experiments, followed by the details on our experiment setup and finally the results obtained from our experiments.

### A. Experiment Dataset

The dataset used in our experiments is built using sentences from the standard *Rottent Tomatoes* (RT) movie review dataset [37]. We obtained the v1.0 balanced binary dataset<sup>1</sup> containing 10,662 reviews. Each movie review presents a users opinion on a movie in about 1-2 sentences. Figure 4 shows a distribution in terms of number of reviews in the original RT dataset for different review lengths (in number of words). About 320 reviews have 5 or lesser words and about 1200 reviews have 10 or lesser words, which can be a severe problem for classical segmentation techniques based sliding windows and statistics of features in these windows. Examples of some reviews from the RT dataset are shown in Table I.

In our experiments we perform a 10-fold cross-validation using the balanced binary dataset of 10,662 reviews. In each fold 90% of the dataset (9596 reviews) are used to build our train set and remaining 10% of the dataset (1066 reviews) are used to form our test set. To build the train set for our task, we randomly sample two reviews from the 9596 reviews allocated for training and concatenate them as a training sample. It will

<sup>1</sup><https://www.cs.cornell.edu/people/pabo/movie-review-data/>

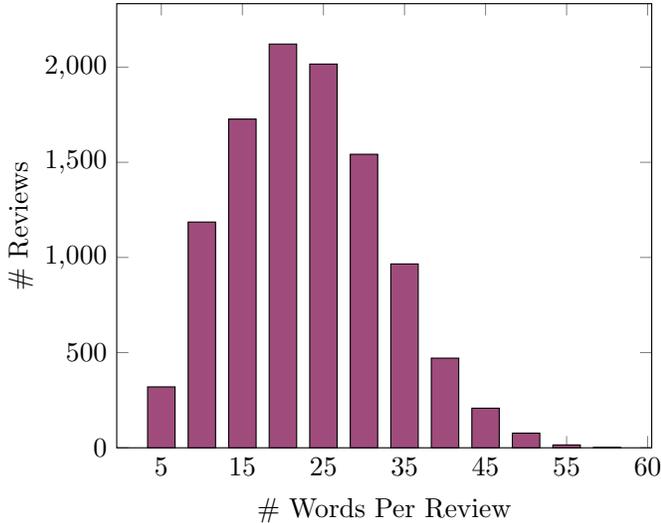


Fig. 4. Distribution of reviews in RT dataset based on review length

TABLE II  
EXAMPLE OF REVIEWS FROM OUR TRAIN AND TEST SET

Sample	Sentiment label	Segment boundary
<i>the jokes are flat, and the action looks fake. truly terrible.</i>	-	11
<i>a big fat pain. few films this year have been as resolute in their emotional nakedness.</i>	-+	5
<i>... there are enough moments of heartbreaking honesty to keep one glued to the screen. an extremely unpleasant film.</i>	+-	16
<i>boisterous, heartfelt comedy. as green-guts monster movies go, it's a beaut.</i>	++	5

be given a sentiment label as -, +-, + or ++ (with - denoting negative and + denoting positive) depending on the sentiment labels of the individual reviews. This label will be used to train the sentiment classification part of the model, as discussed in Section IV. Additionally the length of the first review will be considered as the label for training the segmentation part of the model, as discussed in Section IV. A similar procedure is used to form our test set and its ground truth labels. Table II shows some samples from our training and test set, along with their corresponding classification and segmentation labels.

Figure 5 shows the distribution of segmentation boundaries (length of first segment) for the test set (in fold-0). This distribution is similar for test sets in all the 10 folds of validation. It indicates that the test set has segments of different lengths and difficulties to be identified.

### B. Model Training and Evaluation

As discussed in Section IV, our LSTM-RNN model operates on word embeddings. For our task the word embeddings are initialised with publicly available 300-d (300 dimensional) GloVe word vectors [64], originally trained over the Common

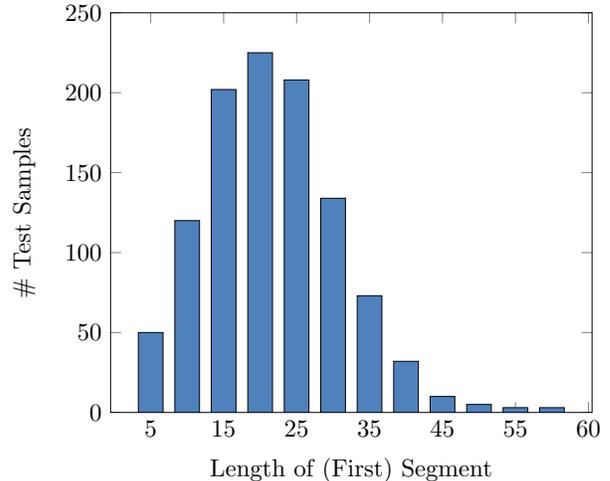


Fig. 5. Distribution of segmentation boundaries (length of first segment) for test set (in fold-0)

Crawl<sup>2</sup>. During model training these word embeddings are treated as model parameters and are updated by using back-propagation, so that the resulting word embedding representations are optimised for our task. Training of all the model parameters is performed with the ADADELTA [65] gradient descent algorithm, which provides an adaptive learning rate. For regularisation of the neural networks and to avoid over-fitting problem, we apply a word dropout [27], [63], with a dropout probability of 0.7. We also apply dropout to the non-sequential units of the LSTM-RNN, as discussed in [66], with a dropout probability of 0.5. Additionally we use an early stopping criterion which keeps a check on the classification error and it stops the model training when the error starts to increase (model over-fitting). The hyper-parameters are chosen based on experiments on the RT sentiment classification task in [27], [63].

As mentioned earlier, we perform a 10-fold cross-validation over the RT dataset. Since our task involves segmentation and classification, we will report *Segmentation Error Rate* (SER) and *Classification Error Rate* (CER), averaged over the 10 folds. The mis-classification of sentiments corresponding to the two segments (-, +, +- or ++) in a test set sample contributes to the classification errors. In segmentation, if the model gives a segmentation such that the segment boundary is more than 3 words away from actual boundary, then the segmentation is treated as erroneous and it contributes to the segmentation error.

### C. Model Performance

Table III presents the segmentation error rate and classification error rate results obtained with our proposed model. We present the results obtained with our model when trained only for segmentation, only for classification and for both segmentation and classification. Moreover we also present the

<sup>2</sup><http://nlp.stanford.edu/projects/glove/>

TABLE III  
MOVIE REVIEW OPINION SEGMENTATION AND CLASSIFICATION RESULTS

Dataset	Model	SER	CER
With full stop and comma	Segmentation Only	0.04	-
	Classification Only	-	0.33
	Segmentation + Classification	0.04	0.31
Without full stop, with comma	Segmentation Only	0.16	-
	Classification Only	-	0.33
	Segmentation + Classification	0.15	0.34
Without full stop and comma	Segmentation Only	0.22	-
	Classification Only	-	0.36
	Segmentation + Classification	0.25	0.35

results when the models are trained and tested on the dataset with full stop marks and commas, after removing full stop marks and after removing both full stop marks and commas. These punctuation marks may carry information for end of segments in the text and it is important to analyse how the segmentation model would perform with and without this information.

It can be observed from Table III, that our proposed model can perform almost perfect segmentation of opinions when the information about sentence boundaries i.e. full stops and other breaks i.e. commas are available. In this case the classification error rate is also lower and it slightly improves when the models are trained with both segmentation and classification cost functions. The segmentation error rate increases when the sentence boundary (full stop) information are not available in train and test, and the error rate reduces further when the commas are also removed. However, an error rate of only 22%, within a strict segmentation criterion of only 3 words, shows that our proposed model can perform even when the full stop and comma punctuation marks are not available. It is also observed that removal of the punctuation marks has a small effect, only 2-5% absolute, on the classification error rate.

We tried to analyse if the errors from our model are due to any particular type of segments or sentiments. In Figure 6 we present the distribution of segmentation errors (in fold-0) for model trained without full stops and commas. The distribution is plotted against length of the first segment. We can see that the distribution of segmentation errors is quite similar to the distribution of the lengths of the first segment in the test set, as shown in Figure 5. This implies that segmentation errors are evenly spread across segments of different lengths, and that the proposed model is not biased towards shorter or longer segments.

We also verified if the segmentation errors are biased towards particular type of sentiment classes (-, +, +- or ++). Again the segmentation errors were evenly distributed across these classes. The fact that the samples with same sentiment segments (- or ++), were not inducing more segmentation errors shows that our model not only learns segmentation of different type of sentiments but also segmentation of different levels of sentiments (of same polarity) and that of different contexts which carry sentiments of same polarity. We further

analysed the errors in sentiment classification and found that they are more or less evenly distributed for the different sentiment classes (-, +, +- and ++), confirming that the model is not biased towards any particular type of sentiment combinations.

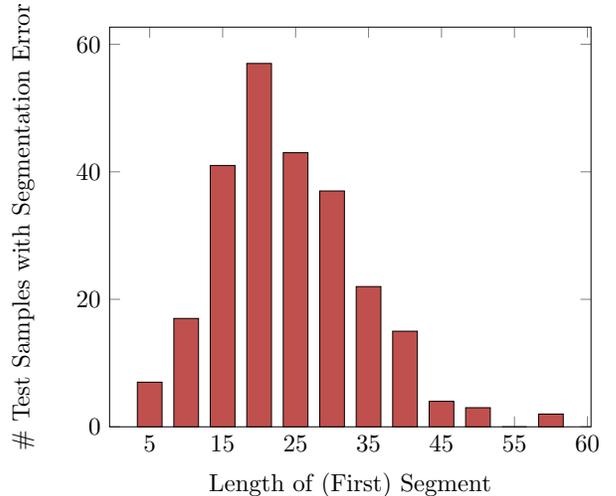


Fig. 6. Distribution of segmentation errors (in fold-0) for model trained without full stop and comma.

## VI. CONCLUSION AND FUTURE WORK

In this paper we proposed a recurrent neural network model with bi-directional LSTM-RNN, to perform joint segmentation and classification of opinions in text. Our proposed model was trained by optimizing network parameters using two cost functions, one for segmentation and other for classification. We introduced a novel method to train bi-directional recurrent neural networks for segmentation. The segmentation cost function compares the sentiment context in the past with that of the future, for every word position in the text, and uses a softmax function to maximise the segmentation accuracy. With experiments on a dataset built from the standard RT movie review dataset, we demonstrated the effectiveness of our proposed model. Our model can perform almost perfect segmentation with knowledge of full stops and comma, which carry information useful for segmentation. We also showed that the model produces good segmentation results even when it is trained and tested on data without full stops and comma.

In our work, we discussed segmentation in context of opinions in text. However, our model readily extends to other sequence segmentation tasks, for example - segmentation of topics in text and automatic speech recognition transcripts. Similarly it can be extended to speech and audio signal segmentation, by operating on automatically trained or pre-computed acoustic features. We are working on these extensions as part of our future work. Moreover, in this paper we focused our work on a sequence with two segments. In future we will extend our model to sequences with more than two segments.

## ACKNOWLEDGMENT

This work is funded by the ContNomina project supported by the French National Research Agency (ANR) under contract ANR-12-BS02-0009. Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria, CNRS, RENATER and other Universities and organisations (<https://www.grid5000.fr>).

## REFERENCES

- [1] P. Nakov, A. Ritter, S. Rosenthal, F. Sebastiani, and V. Stoyanov, "Semeval-2016 task 4: Sentiment analysis in twitter," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 1–18.
- [2] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. Loukachevitch, E. Kotelnikov, N. Bel, S. M. Jiménez-Zafra, and G. Eryiğit, "Semeval-2016 task 5: Aspect based sentiment analysis," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 19–30.
- [3] S. Kiritchenko, S. Mohammad, and M. Salameh, "Semeval-2016 task 7: Determining sentiment intensity of english and arabic phrases," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 42–51.
- [4] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 31–41.
- [5] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and Trends in Information Retrieval*, vol. 2, no. 12, pp. 1–135, 2008.
- [6] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093 – 1113, 2014.
- [7] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '04. New York, NY, USA: ACM, 2004, pp. 168–177.
- [8] A. Ghose, P. G. Ipeirotis, and A. Sundararajan, "Opinion Mining using Econometrics: A Case Study on Reputation Systems," in *ACL*. The Association for Computer Linguistics, 2007.
- [9] H. Saif, M. Fernandez, Y. He, and H. Alani, "Evaluation datasets for twitter sentiment analysis a survey and a new dataset, the sts-gold," in *Interantional Workshop on Emotion and Sentiment in Social and Expressive Media: Approaches and Perspectives from AI (ESSEM 2013)*, 2013.
- [10] W. Gryc and K. Moilanen, "Leveraging textual sentiment analysis with social network modelling," *From Text to Political Positions: Text analysis across disciplines*, vol. 55, p. 47, 2014.
- [11] A. Ceron, L. Curini, and S. M. Iacus, "Using sentiment analysis to monitor electoral campaigns: Method matters-evidence from the united states and italy," *Soc. Sci. Comput. Rev.*, vol. 33, no. 1, pp. 3–20, Feb. 2015.
- [12] H. Wang, D. Can, A. Kazemzadeh, F. Bar, and S. Narayanan, "A system for real-time twitter sentiment analysis of 2012 u.s. presidential election cycle," in *Proceedings of the ACL 2012 System Demonstrations*, ser. ACL '12. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 115–120.
- [13] Y. Mejova, P. Srinivasan, and B. Boynton, "Gop primary season on twitter: "popular" political sentiment in social media," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, ser. WSDM '13. New York, NY, USA: ACM, 2013, pp. 517–526.
- [14] Y. He, H. Saif, Z. Wei, and K. fai Wong, "Quantising opinions for political tweets analysis," in *Eight International Conference on Language Resources and Evaluation (LREC'12)*, European Language Resources Association (ELRA, 2012.
- [15] L.-P. Morency, R. Mihalcea, and P. Doshi, "Towards multimodal sentiment analysis: Harvesting opinions from the web," in *Proceedings of the 13th International Conference on Multimodal Interfaces*, ser. ICMI '11. New York, NY, USA: ACM, 2011, pp. 169–176.
- [16] S. Poria, E. Cambria, N. Howard, G.-B. Huang, and A. Hussain, "Fusing audio, visual and textual clues for sentiment analysis from multimodal content," *Neurocomput.*, vol. 174, no. PA, pp. 50–59, Jan. 2016.
- [17] M. Wollmer, F. Weninger, T. Knaup, B. Schuller, C. Sun, K. Sagae, and L.-P. Morency, "Youtube movie reviews: Sentiment analysis in an audio-visual context," *IEEE Intelligent Systems*, vol. 28, no. 3, pp. 46–53, 2013.
- [18] J. G. Ellis, B. Jou, and S.-F. Chang, "Why we watch the news: A dataset for exploring sentiment in broadcast video news," in *Proceedings of the 16th International Conference on Multimodal Interaction*, ser. ICMI '14. New York, NY, USA: ACM, 2014, pp. 104–111.
- [19] V. P. Rosas, R. Mihalcea, and L.-P. Morency, "Multimodal Sentiment Analysis of Spanish Online Videos," *IEEE Intelligent Systems*, vol. 28, no. 3, pp. 38 – 45, Jan. 2013.
- [20] G. Mishne, D. Carmel, R. Hoory, A. Roytman, and A. Soffer, "Automatic analysis of call-center conversations," in *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, ser. CIKM '05. New York, NY, USA: ACM, 2005, pp. 453–459.
- [21] C. Clavel, G. Adda, F. Cailliau, M. Garnier-Rizet, A. Cavet, G. Chapuis, S. Courcinous, C. Danesi, A.-L. Daquo, M. Deldossi, S. Guillemin-Lanne, M. Seizou, and P. Suignard, "Spontaneous speech and opinion detection: mining call-centre transcripts," *Language Resources and Evaluation*, vol. 47, no. 4, pp. 1089–1125, 2013.
- [22] S. Ezzat, N. E. Gayar, and M. M. Ghanem, "Sentiment analysis of call centre audio conversations using text classification," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 4, pp. 619–627, 2012.
- [23] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, ser. EMNLP '02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 79–86.
- [24] K. Schouten and F. Frasincar, "Survey on aspect-level sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 813–830, March 2016.
- [25] Y. Goldberg, "A primer on neural network models for natural language processing," *CoRR*, vol. abs/1510.00726, 2015.
- [26] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, 2014, pp. 1188–1196.
- [27] M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H. Daumé III, "Deep unordered composition rivals syntactic methods for text classification," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 1681–1691.
- [28] J. Nam, J. Kim, E. Loza Mencía, I. Gurevych, and J. Fürnkranz, "Large-scale multi-label text classification - revisiting neural networks," in *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD-14)*, Part 2, ser. Lecture Notes in Computer Science, T. Calders, F. Esposito, E. Hüllermeier, and R. Meo, Eds., vol. 8725. Springer Berlin Heidelberg, Sep. 2014, pp. 437–452.
- [29] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, October 2014, pp. 1746–1751.
- [30] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolutional neural networks," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, May–June 2015, pp. 103–112.

- [31] P. Wang, J. Xu, B. Xu, C. Liu, H. Zhang, F. Wang, and H. Hao, "Semantic clustering and convolutional neural network for short text categorization," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 352–357.
- [32] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA: Association for Computational Linguistics, October 2013, pp. 1631–1642.
- [33] K. M. Hermann and P. Blunsom, "The Role of Syntax in Vector Space Models of Compositional Semantics," in *Proceedings of ACL*, Aug. 2013.
- [34] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent twitter sentiment classification," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, 2014, pp. 49–54.
- [35] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 1556–1566.
- [36] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 3061–3069.
- [37] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ser. ACL '05. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 115–124.
- [38] A. Sharma and S. Dey, "An artificial neural network based approach for sentiment analysis of opinionated text," in *Proceedings of the 2012 ACM Research in Applied Computation Symposium*, ser. RACS '12. New York, NY, USA: ACM, 2012, pp. 37–42.
- [39] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 142–150.
- [40] D. Tang, B. Qin, and T. Liu, "Deep learning for sentiment analysis: Successful approaches and future challenges," *Wiley Int. Rev. Data Min. and Knowl. Disc.*, vol. 5, no. 6, pp. 292–303, Nov. 2015.
- [41] S. Poria, E. Cambria, and A. Gelbukh, "Aspect extraction for opinion mining with a deep convolutional neural network," *Knowledge-Based Systems*, vol. 108, pp. 42 – 49, 2016, new Avenues in Knowledge Bases for Natural Language Processing.
- [42] T. H. Nguyen and K. Shirai, "Phrasernn: Phrase recursive neural network for aspect-based sentiment analysis," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, September 2015, pp. 2509–2514.
- [43] O. Irsoy and C. Cardie, "Opinion mining with deep recurrent neural networks," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, October 2014, pp. 720–728.
- [44] P. Liu, S. Joty, and H. Meng, "Fine-grained opinion mining with recurrent neural networks and word embeddings," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, September 2015, pp. 1433–1443.
- [45] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, September 2015, pp. 1422–1432.
- [46] M. A. Hearst, "Texttiling: Segmenting text into multi-paragraph subtopic passages," *Comput. Linguist.*, vol. 23, no. 1, pp. 33–64, Mar. 1997.
- [47] F. Y. Y. Choi, "Advances in domain independent linear text segmentation," in *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, ser. NAACL 2000. Stroudsburg, PA, USA: Association for Computational Linguistics, 2000, pp. 26–33.
- [48] H. Misra, F. Yvon, J. M. Jose, and O. Cappe, "Text segmentation via topic modeling: An analytical study," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ser. CIKM '09. New York, NY, USA: ACM, 2009, pp. 1553–1556.
- [49] M. Riedl and C. Biemann, "Text segmentation with topic models," *Journal for Language Technology and Computational Linguistics*, vol. 27, no. 1, pp. 47–69, 2012.
- [50] H. Misra, F. Yvon, O. Capp, and J. Jose, "Text segmentation: A topic modeling perspective," *Information Processing & Management*, vol. 47, no. 4, pp. 528 – 544, 2011.
- [51] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *EMNLP*, vol. 14, 2014, pp. 1532–43.
- [52] A. A. Alemi and P. Ginsparg, "Text segmentation based on semantic word embeddings," *CoRR*, vol. abs/1503.05543, 2015.
- [53] D. Tang, B. Qin, F. Wei, L. Dong, T. Liu, and M. Zhou, "A joint segmentation and classification framework for sentence level sentiment classification," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 23, no. 11, pp. 1750–1761, Nov. 2015.
- [54] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179 – 211, 1990.
- [55] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural Networks: Tricks of the Trade. This Book is an Outgrowth of a 1996 NIPS Workshop*. London, UK, UK: Springer-Verlag, 1998, pp. 9–50.
- [56] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," *CoRR*, vol. abs/1206.5533, 2012.
- [57] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *ICML (3)*, ser. JMLR Workshop and Conference Proceedings, vol. 28. JMLR.org, 2013, pp. 1310–1318.
- [58] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [59] W. Commons, "Long short term memory," [https://commons.wikimedia.org/wiki/File:Long\\_Short\\_Term\\_Memory.png](https://commons.wikimedia.org/wiki/File:Long_Short_Term_Memory.png), 2015.
- [60] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," *Trans. Sig. Proc.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997. [Online]. Available: <http://dx.doi.org/10.1109/78.650093>
- [61] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm networks," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 4, July 2005, pp. 2047–2052 vol. 4.
- [62] M. Sundermeyer, T. Alkhoul, J. Wuebker, and H. Ney, "Translation modeling with bidirectional recurrent neural networks," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, October 2014, pp. 14–25.
- [63] I. Sheikh, I. Illina, D. Fohr, and G. Linarès, "Learning word importance with the neural bag-of-words model," in *Proceedings of the 1st Workshop on Representation Learning for NLP*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 222–229.
- [64] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.
- [65] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012.
- [66] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *CoRR*, vol. abs/1409.2329, 2014.