Computers should pay better Attention to Documents!

1st Vinayak Sengupta
Rochester Institute of Technology
vs4016@rit.edu

***Abstract*—To classify documents, we propose hierarchical attention network(HAN). It possesses two important characteristics: (i) its hierarchical organization mirrors the structure of documents; (ii) it possesses two levels of attention mechanisms applied at the sentence and word levels, allowing it to pay differential attention to more and less important content when constructing the document representation. A previous study regarding textual classification using HANs will be examined a little further within this project. Our approach is designed to improve performance by modernizing it and simplifying it based on what we believe is necessary. Studies conducted on a particular set of IMDB reviews, on which previous research had not been successful in achieving a high model accuracy and we are seeking to outperform them. The recommendation architecture significantly outperforms the previous method in the large-scale text classification task. The model selects words and sentences that provide qualitative information (visualization of the attention layers). The previous implementation on the same data received a accuracy of 49.4%, while we achieve a sentence level accuracy of 88.4% and a word level accuracy of 84.4%.**

I. Introduction

Natural Language Processing includes the task of text classification, which entails assigning labels to words. It has broad applications including topic labeling [2] and sentiment classification [3]. Various deep learning approaches have been developed recently to learn text representations, including convolutional neural networks and recurrent neural networks based on long short-term memory. This model implies that not all data in a document is equally useful for answering a query, and that to determine which sections are useful, it is necessary to take into account their interactions rather than looking at their presence alone.

Two basic insights about document structure can be captured by the Hierarchical Attention Network(HAN). First, the document representation is similarly created by first constructing representations of sentences, then aggregating those into a document representation (words form sentences, sentences form a document) so that one can understand the hierarchical structure of the document. Second, quite often, different words and sentences within a document hold differential information value. On top of that, different words and sentences may have different meanings depending on the context.

An architecture diagram of the HAN is given in figure 1. The architecture is composed of several parts, including a word sequence encoder, a word-level attention layer, a sentence encoder, and a sentence-level attention layer.

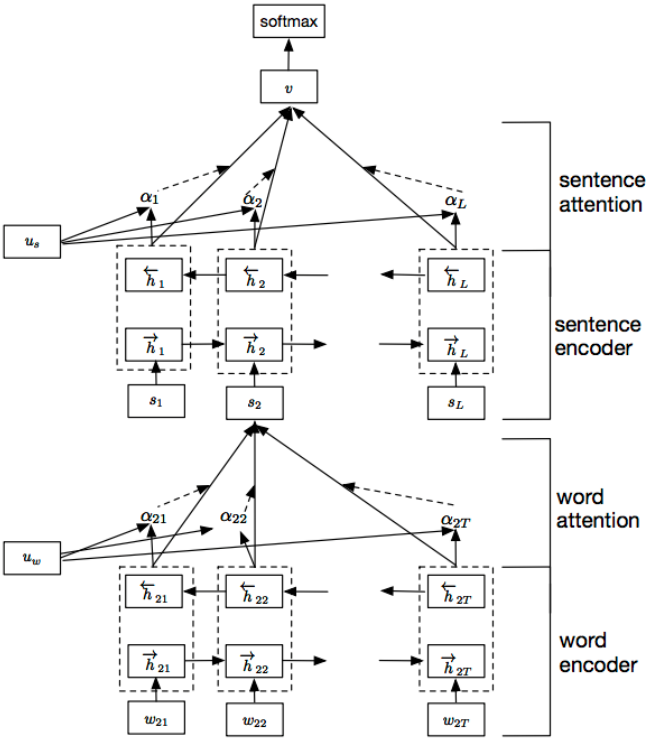

Fig. 1. Hierarchical Attention Network

We use context rather than simply filtering for (sequences of) tokens out of context to discover when a sequence is relevant as opposed to previous work that only looked at tokens in isolation. Since the work we compare our model to had a low model accuracy, we looked at IMDB reviews to determine how the model performed against the competition. Our model outperformed previous approaches by a significant amount.

II. Data

On the large scale document classification data set from IMDB, we evaluate the effectiveness of our model. Sentiment analysis was performed on 50,000 reviews selected specifically from IMDB for this data set. IMDB ratings have a binary sentiment system, meaning ratings of 5 and better

result in a score of 0, whereas ratings over 7 result in a score of 1. In addition, the training set includes another 50,000 IMDB reviews without any rating labels. The training set of 25,000 reviews labeled in green does not include any of the same movies as the test set. Below is the file as well as the data-field descriptions:

File Description:

- **labeledTrainData** - 25,000 rows of tab-delimited ratings, each with its own identifier, sentiment, and text. The labeled training set.
- **testData** - The tab-delimited file contains 25,000 rows that are used to predict the sentiment of every review. The header row is followed by 25,000 rows of text and an id. The test set.
- **unlabeledTrainData** - This file contains 50,000 rows of text and an id for each review. The tab-delimited file has a header row following 50,000 rows with no labels.

Data-field Description:

- **id** - Each review has its own unique ID.
- **sentiment** - Positive reviews receive 1 and negative reviews receive 0.
- **review** - Annotated text of the review

III. Methods

The Hierarchical Attention Network will be implemented as a base line of a Hierarchical LSTM network. In my previous posts, I constructed the data input as 2D, but now I have to devise a 3D input. Therefore, the input tensor would be the number of reviews per batch, the number of sentences, and the number of words of each sentence.

After that, the hierarchical input layers can be constructed using Keras' function TimeDistributed. It is the TimeDistributed layer which enables each sentence in a document to be encoded. The model's final output is a sigmoid function, which predicts 1 for positive sentiment and 0 for negative sentiment[1]. Compared with one level of LSTM, the training time now is much faster. In this endevour we have implemented the classification model using feed-forward networks with attention [4]. To implement the attention layer, we built a custom Keras layer. Also, the output from GRU is fed to a dense layer before being fed into the attention layer. In the following implementation, the attention network is divided into two layers, one at sentence level and the other at review level.

Figure 2, is a representation of how our feed-forward attention model works. The vectors in the hidden state h_t are inputs to a function $a(h_t)$,this produces a probability vector α. The vector c is then calculated as a weighted average of h_t, with the α providing the weights.

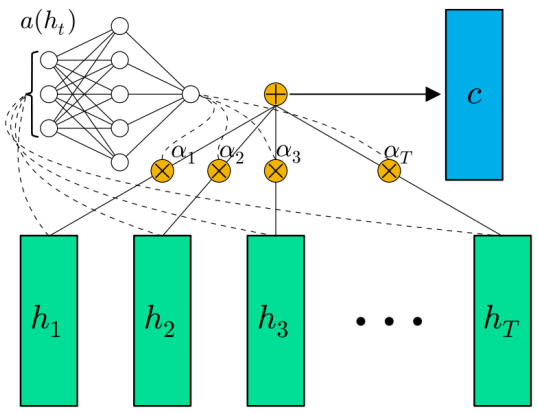

Fig. 2. Hierarchical Attention Network

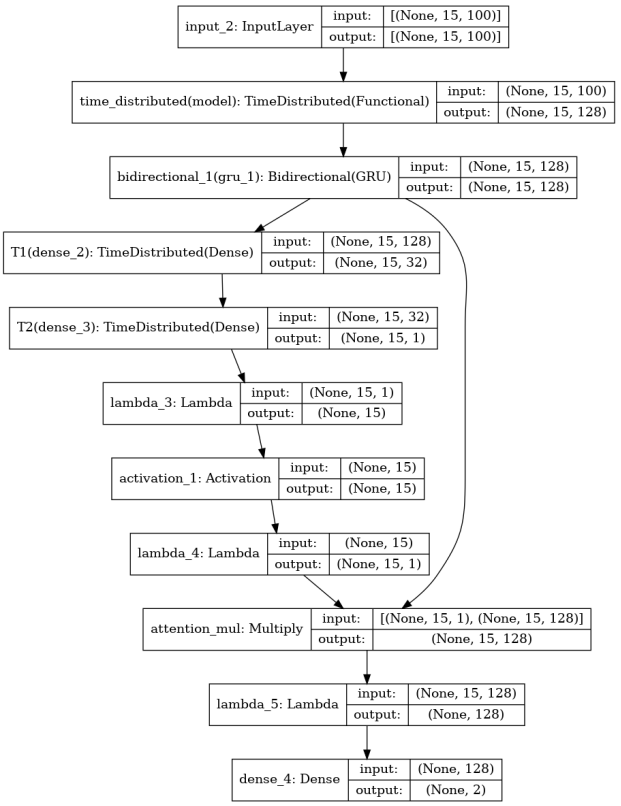

Fig. 3. Final Hierarchical Attention Network with GRU architecture

The output from the GRU is then added to a dense layer before being fed into our attention layer, which has two layers, one on a sentence level and the other on a review level.

The final model architecture is displayed in Figure 3.

[1]https://offbit.github.io/how-to-read/

Author	Model	Data Set	Accuracy (in %)
Yang et al.	HN-ATT	IMDB	49.4
Vinayak (me)	HN-GRU (Document)	IMDB	88.4
Vinayak (me)	HN-GRU (Sentence)	IMDB	84.4

TABLE I
MODEL RESULTS COMPARISON

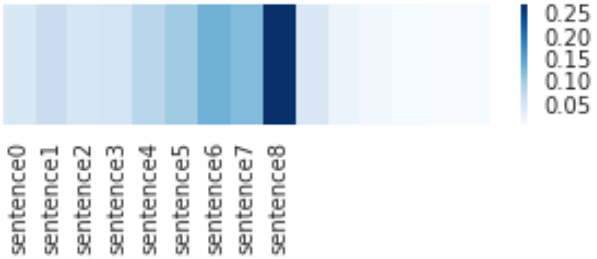

Fig. 4. Sentence attention weights for document classification

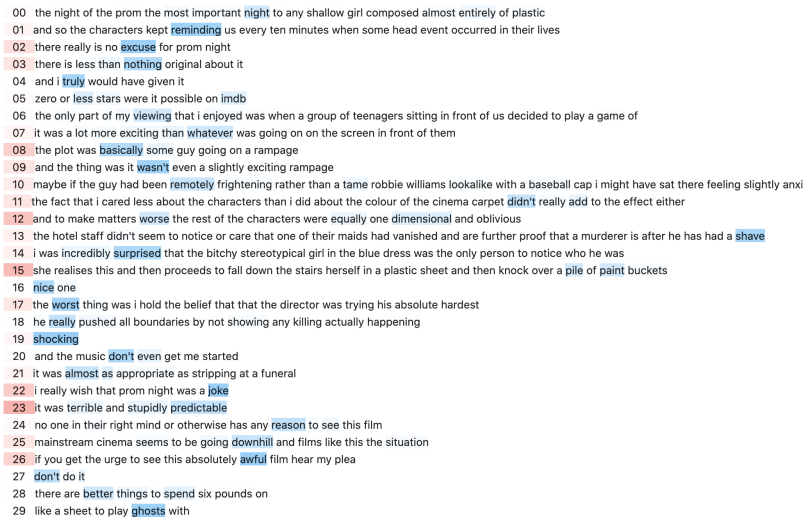

Fig. 5. Sentence attention weights, color-coded

IV. RESULTS & DISCUSSION

A. *Metric Comparison*

We want to have a fair comparison of our work against that of the previous work by Yang et al. [1] that we are working to improve. Hence, for that to take place, we have utilised the same anotated dataset by IMDB that the previous authors have used. We will also be comparing the 'Accuracy' metric as that is what the paper did as well. The results can be seen in the given table 1.

B. *Weights Visualization*

On getting results, we ofcourse want to analyse how they look, so for this we conduct the attention weights visualization. The principle idea surrounding this task is performing a forward pass.

We start by defining a K.function (from Keras) and derive the GRU layer or whichever layer, that is the output before Attention input. We repeat this process for the attention weights calculation as well. The weights dimension that we have considered is 1000. From the weights outputwe can then extract n number of top words that we are interested in.

Figure 4, showcases the sentence wise attention allocation by the model towards document classification. Figure 5, visualises the word wise attention allocation by the model towards sentence classification. Figure 6, visualises the same as 5, in this case the blue shade refers to words with high attention weight, while red refers to words with low attention weights.

C. *Inference from Results*

*1) **Data**:* One of the most immediate differences in the approaches was the dataset itself. The authors utilised an IMDB dataset that was not very rich in quantity with only 25000 rows. While we utilised a dataset with 50000 rows. This two times difference in data quantity makes a considerable difference, as the model has much more information than before to learn weights by and generalise better.

*2) **Modelling**:* There are a few modelling approach differences, which we feel has attributed towards our substantially better performance. Firstly, the implementation of the feed-forward with the attention network overcame some of the problems, including time complexity which was faced with the Long-Short-Term-memory network.

Second, the addition of a dense layer, which took the output from the GRU before feeding into input layer, was a wise decision as that implicated in the implementation of two layers attention (one at sentence level and one at document level), which has helped immensely in weight learning towards specific inputs. This specialising of 2 separate inputs has helped since it makes the model concentrate on specific areas of the document and learn their attention weights, and then using those sentence weights to further classify by word attention.

V. CONCLUSION

We set out to an endeavour to try and improve upon a pioneering research in the research of automated document classification using natural language techniques along with machine learning architectures. We have not only been able to successfully build on top of existing work, but also have been successful in taking their results further and improved them considerably.

We have been able to gain sentence-wise attention generation to classify which sentences matter when a computer classifies a document. We have also gained word-wise attention weights to classify which words matter to a computer

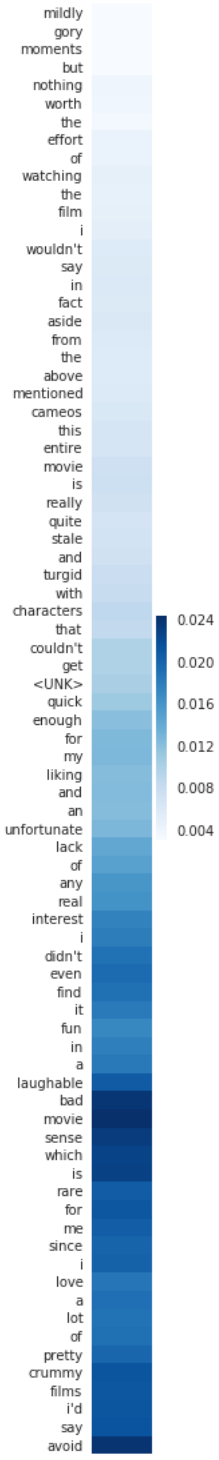

Fig. 6. Word attention weights for sentence classification

when its trying to classify sentences. A hierarchical attention network, which combines these two models, could potentially be the best method of text classification since it describes the importance of words and sentences.

REFERENCES

[1] Hierarchical attention networks for document classification Z Yang, D Yang, C Dyer, X He, A Smola, E Hovy - Proceedings of the 2016 conference of the North ..., 2016

[2] Wang, Sida & Manning, Christopher. (2012). Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. 90-94.

[3] Maas, Andrew Daly, Raymond Pham, Peter Huang, Dan Ng, Andrew Potts, Christopher. (2011). Learning Word Vectors for Sentiment Analysis. 142-150.

[4] Raffel, Colin Ellis, Daniel. (2015). Feed-Forward Networks with Attention Can Solve Some Long-Term Memory Problems.