Towards Abstraction from Extraction: Multiple Timescale Gated Recurrent Unit for Summarization

Minsoo Kim
School of Electronics Engineering
Kyungpook National University
Daegu, South Korea
`minsoo9574@gmail.com`

Moirangthem Dennis Singh
School of Electronics Engineering
Kyungpook National University
Daegu, South Korea
`mdennissingh@gmail.com`

Minho Lee
School of Electronics Engineering
Kyungpook National University
Daegu, South Korea
`mholee@gmail.com`

Abstract

In this work, we introduce temporal hierarchies to the sequence to sequence (seq2seq) model to tackle the problem of abstractive summarization of scientific articles. The proposed Multiple Timescale model of the Gated Recurrent Unit (MT-GRU) is implemented in the encoder-decoder setting to better deal with the presence of multiple compositionalities in larger texts. The proposed model is compared to the conventional RNN encoder-decoder, and the results demonstrate that our model trains faster and shows significant performance gains. The results also show that the temporal hierarchies help improve the ability of seq2seq models to capture compositionalities better without the presence of highly complex architectural hierarchies.

1 Introduction and Related Works

Summarization has been extensively researched over the past several decades. Jones (2007) and Nenkova et al. (2011) offer excellent overviews of the field. Broadly, summarization methods can be categorized into extractive approaches and abstractive approaches (Hahn and Mani, 2000), based on the type of computational task. Extractive summarization is a selection problem, while abstractive summarization requires a deeper semantic and discourse understanding of the text, as well as a novel text generation process. Extractive summarization has been the focus in the past, but abstractive summarization remains a challenge.

Recently, sequence-to-sequence (seq2seq) recurrent neural networks (RNNs) have seen wide application in a number of tasks. Such RNN encoder-decoders (Cho et al., 2014; Bahdanau et al., 2014) combine a representation learning encoder and a language modeling decoder to perform mappings between two sequences. Similarly, recent works have proposed to cast summarization as a mapping problem between an input sequence and a summary sequence. Recent successes such as Rush et al. (2015);Nallapati et al. (2016) have shown that the RNN encoder-decoder performs remarkably well in summarizing short text. Such seq2seq approaches offer a fully data-driven solution to both semantic and discourse understanding and text generation.

While seq2seq presents a promising way forward for abstractive summarization, extrapolating the methodology to other tasks, such as the summarization of a scientific article, is not trivial. A number of practical and theoretical concerns arise: 1) We cannot simply train RNN encoder-decoders on entire articles: For the memory capacity of current GPUs, scientific articles are too long to be processed whole via RNNs. 2) Moving from one or two sentences, to several sentences or several paragraphs, introduces additional levels of compositionality and richer discourse structure. How can we improve the conventional RNN encoder-decoder to better capture these? 3) Deep learning approaches depend heavily on good quality, large-scale datasets. Collecting source-summary data pairs is difficult, and datasets are scarce outside of the newswire domain.

In this paper, we present a first, intermediate step towards end-to-end abstractive summarization of scientific articles. Our aim is to extend seq2seq based summarization to larger text with a more complex summarization task. To ad-

dress each of the issues above, 1) We propose a paragraph-wise summarization system, which is trained via paragraph-salient sentence pairs. We use Term Frequency-Inverse Document Frequency (TF-IDF) (Luhn, 1958; Jones, 1972) scores to extract a salient sentence from each paragraph. 2) We introduce a novel model, Multiple Timescale Gated Recurrent Unit (MTGRU), which adds a temporal hierarchy component that serves to handle multiple levels of compositionality. This is inspired by an analogous concept of temporal hierarchical organization found in the human brain, and is implemented by modulating different layers of the multilayer RNN with different timescales (Yamashita and Tani, 2008). We demonstrate that our model is capable of understanding the semantics of a multi-sentence source text and knowing what is important about it, which is the first necessary step towards abstractive summarization. 3) We build a new dataset of Computer Science (CS) articles from ArXiv.org, extracting their Introductions from the LaTeX source files. The Introductions are decomposed into paragraphs, each paragraph acting as a natural unit of discourse.

Finally, we concatenate the generated summary of each paragraph to create a non-expert summary of the article's Introduction, and evaluate our results against the actual Abstract. We show that our model is capable of summarizing multiple sentences to its most salient part on unseen data, further supporting the larger view of summarization as a seq2seq mapping task. We demonstrate that our MTGRU model satisfies some of the major requirements of an abstractive summarization system. We also report that MTGRU has the capability of reducing training time significantly compared to the conventional RNN encoder-decoder.

The paper is structured as follows: Section 2 describes the proposed model in detail. In Section 3, we report the results of our experiments and show the generated summary samples. In Section 4 we analyze the results of our model and comment on future work.

2 Proposed Model

In this section we discuss the background related to our model, and describe in detail the newly developed architecture and its application to summarization.
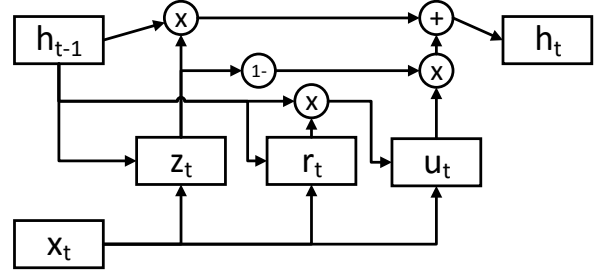

Figure 1: A gated recurrent unit.

2.1 Background

The principle of compositionality defines the meaning conveyed by a linguistic expression as a function of the syntactic combination of its constituent units. In other words, the meaning of a sentence is determined by the way its words are combined with each other. In multi-sentence text, sentence-level compositionality (the way sentences are combined with one another) is an additional function which will add meaning to the overall text. When dealing with such larger texts, compositionality at the sentence and even paragraph levels should be considered, in order to capture the text meaning completely. An approach explored in recent literature is to create dedicated architectures in a hierarchical fashion to capture subsequent levels of compositionality: Li et al. (2015) and Nallapati et al. (2016) build dedicated word and sentence level RNN architectures to capture compositionality at different levels of text-units, leading to improvements in performance.

However, architectural modifications to the RNN encoder-decoder such as these suffer from the drawback of a major increase in both training time and memory usage. Therefore, we propose an alternative enhancement to the architecture that will improve performance with no such overhead. We draw our inspiration from neuroscience, where it has been shown that functional differentiation occurs naturally in the human brain, giving rise to temporal hierarchies (Meunier et al., 2010; Botvinick, 2007). It has been well documented that neurons can hierarchically organize themselves into layers with different adaptation rates to stimuli. The quintessential example of this phenomenon is the auditory system, in which syllable level information in a short time window is integrated into word level information over a longer time window, and so on. Previous works have applied this concept to RNNs in movement tracking (Paine and Tani, 2004) and speech recog-
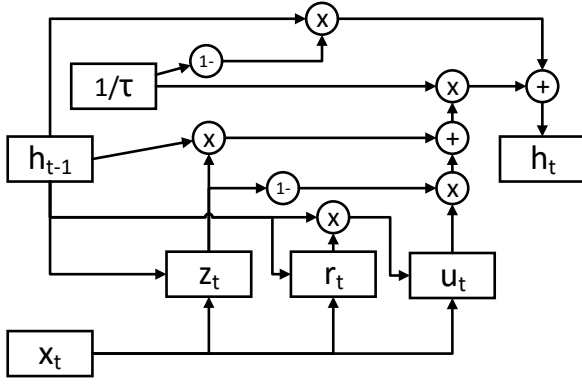
Figure 2: Proposed multiple timescale gated recurrent unit.

nition (Heinrich et al., 2012).

2.2 Multiple Timescale Gated Recurrent Unit

Our proposed Multiple Timescale Gated Recurrent Unit (MTGRU) model applies the temporal hierarchy concept to the problem of seq2seq text summarization, in the framework of the RNN encoder-decoder. Previous works such as (Yamashita and Tani, 2008)'s Multiple Timescale Recurrent Neural Network (MTRNN) have employed temporal hierarchy in motion prediction. However, MTRNN is prone to the same problems present in the RNN, such as difficulty in capturing long-term dependencies and vanishing gradient problem (Hochreiter et al., 2001). Long Short Term Memory network (Hochreiter et al., 2001) utilizes a complex gating architecture to aid the learning of long-term dependencies and has been shown to perform much better than the RNN in tasks with long-term temporal dependencies such as machine translation (Sutskever et al., 2014). Gated Recurrent Unit (GRU) (Cho et al., 2014), which has been proven to be comparable to LSTM (Chung et al., 2014), has a similar complex gating architecture, but requires less memory. The standard GRU architecture is shown in Fig. 1.

Because seq2seq summarization involves potentially many long-range temporal dependencies, our model applies temporal hierarchy to the GRU. We apply a timescale constant at the end of a GRU, essentially adding another constant gating unit that modulates the mixture of past and current hidden states. The reset gate r_t, update gate z_t, and the candidate activation u_t are computed similarly to that of the original GRU as shown in Eq.(1).

$$r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1})$$
$$z_t = \sigma(W_{xz}x_t + W_{hz}h_{t-1}) \quad (1)$$
$$u_t = \tanh(W_{xu}x_t + W_{hu}(r_t \odot h_{t-1}))$$
$$h_t = ((1 - z_t)h_{t-1} + z_t u_t)\frac{1}{\tau} + (1 - \frac{1}{\tau})h_{t-1} \quad (2)$$

The time constant τ added to the activation h_t of the MTGRU is shown in Eq.(2). τ is used to control the timescale of each GRU cell. Larger τ meaning slower cell outputs but it makes the cell focus on the slow features of a dynamic sequence input. The proposed MTGRU model is illustrated in Fig. 2. The conventional GRU will be a special case of MTGRU where $\tau = 1$, where no attempt is made to organize layers into different timescales.

$$\frac{\delta E}{\delta h_{t-1}} = \frac{1}{\tau}[\frac{\delta E}{\delta h_t} \odot (u_t - h_{t-1}) \odot \sigma'(z_t)W_{zh}]$$
$$+ \frac{1}{\tau}[((\frac{\delta E}{\delta h_t} \odot z_t \odot \tanh'(u_t))W_{uh}) \odot r_t]$$
$$+ \frac{1}{\tau}[(((\frac{\delta E}{\delta h_t} \odot z_t \odot \tanh'(u_t))W_{uh})$$
$$\odot \sigma'(r_t) \odot h_{t-1})W_{rh}]$$
$$+ \frac{1}{\tau}[\frac{\delta E}{\delta h_t} \odot (1 - z_t)] + (1 - \frac{1}{\tau})\frac{\delta E}{\delta h_t} \quad (3)$$

Eq. (3) shows the learning algorithm derived for the MTGRU according to the defined forward process and the back propagation through time rules. $\frac{\delta E}{\delta h_{t-1}}$ is the error of the cell outputs at time $t - 1$ and $\frac{\delta E}{\delta h_t}$ is the current gradient of the cell outputs. Different timescale constants are set for each layer where larger τ means slower context units and $\tau = 1$ defines the default or the input timescale. Based on our hypothesis that later layers should learn features that operate over slower timescales, we set larger τ as we go up the layers.

In this application, the question is whether the word sequences being analyzed by the RNN possess information that operates over different temporal hierarchies, as they do in the case of the continuous audio signals received by the human auditory system. We hypothesize that they do, and that word level, clause level, and sentence level compositionalities are strong candidates. In this light, the multiple timescale modification functions as a way to explicitly guide each layer of the neural network to facilitate the learning of features operating over increasingly slower timescales, corre-

RNN Type	Layers	Hidden Units
GRU	4	1792
MTGRU	4	1792

Table 1: Network Parameters for each model.

sponding to subsequent levels in the compositional hierarchy.

2.3 Summarization

To apply our newly proposed multiple timescale model to summarization, we build a new dataset of academic articles. We collect LaTeX source files of articles in the CS.{CL,CV,LG,NE} domains from the arXiv preprint server, extracting their Introductions and Abstracts. We decompose the Introduction into paragraphs, and pair each paragraph with its most salient sentence as the target summary. These target summaries are generated using the widely adopted TF-IDF scoring. Fig. 3 shows the structure of our summarization model.

Our dataset contains rich compositionality and longer text sequences, increasing the complexity of the summarization problem. The temporal hierarchy function has the biggest impact when complex compositional hierarchies exist in the input data. Hence, the multiple timescale concept will play a bigger role in our context compared to previous summarization tasks such as Rush et al. (2015).

The model using MTGRU is trained using these paragraphs and their targets. The generated summaries of each Introduction is evaluated using the Abstracts of the collected articles. We chose the Abstracts as gold summaries, because they usually contain important discourse structures such as goal, related works, methods, and results, making them good baseline summaries. To test the effectiveness of the proposed method, we compare it with the conventional RNN encoder-decoder in terms of training speed and performance.

3 Experiments and Results

We trained two seq2seq models, the first model using the conventional GRU in the RNN encoder decoder, and the second model using the newly proposed MTGRU. Both models are trained using the same hyperparamenter settings with the optimal configuration which fits our existing hardware capability.

Following Sutskever et al. (2014), the inputs are divided into multiple buckets. Both GRU and MT-
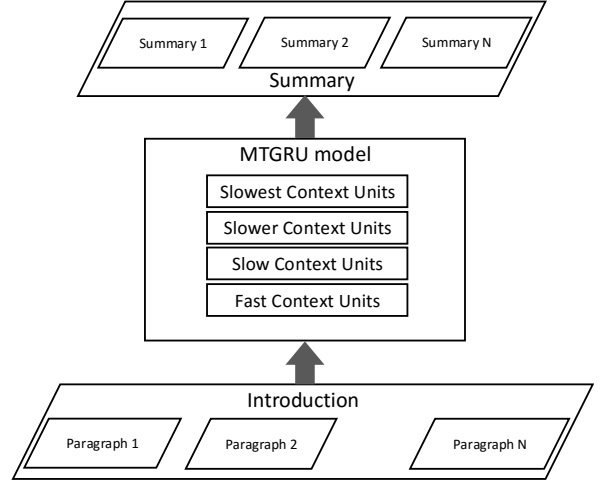

Figure 3: Paragraph level approach to summarization.

Steps	RNN	Train Perplexity	Test Perplexity
74750	GRU	6.8	29.72
74750	MTGRU	5.87	18.53

Table 2: Training results of the Models.

GRU models consist of 4 layers and 1792 hidden units. As our models take longer input and target sequence sizes, the hidden units size and number of layers are limited. An embedding size of 512 was used for both networks. The timescale constant τ for each layer is set to $1, 1.25, 1.5, 1.7$, respectively. The models are trained on 110k text-summary pairs. The source text are the paragraphs extracted from the introduction of academic articles and the targets are the most salient sentence extracted from the paragraphs using TF-IDF scores. For comparison of the training speed of the models, Fig. 4 shows the plot of the training curve until the train perplexity reaches 9.5. Both of the models are trained using 2 Nvidia Ge-Force GTX Titan X GPUs which takes roughly 4 days and 3 days respectively. During test, greedy decoding was used to generate the most likely output given a source Introduction.

For evaluation, we adopt the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics (Lin, 2004) proposed by Lin and Hovy (2003). ROUGE is a recall-oriented measure to score system summaries which is proven to have a strong correlation with human evaluations. It measures

Evaluation Metric	Recall	Precision	F–Score
ROUGE-1	0.48135	0.59030	0.50835
ROUGE-2	0.32399	0.39505	0.34089
ROUGE-L	0.46588	0.57218	0.49234

Table 3: ROUGE scores of GRU Model

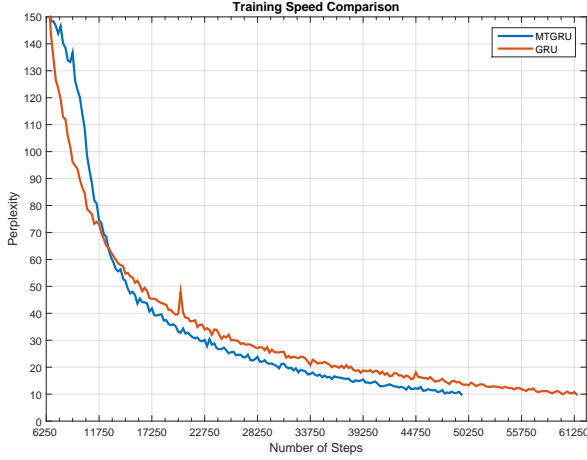| Evaluation Metric	Recall	Precision	F–Score
ROUGE-1	0.50901	0.61571	0.53870
ROUGE-2	0.34148	0.40824	0.35925
ROUGE-L	0.49406	0.59830	0.52318

Table 4: ROUGE scores of MTGRU Model

Figure 4: Comparison of Training Speed between GRU and MTGRU.

the n-gram recall between the candidate summary and gold summaries. In this work, we only have one gold summary which is the Abstract of an article, thus the ROUGE score is calculated as given in Li et al. (2015). ROUGE-1, ROUGE-2 and ROUGE-L are used to report the performance of the models. For the performance evaluation, both the models are trained up to 74750 steps where the training perplexity of GRU and MTGRU are shown in Table 2. This step was chosen as the early stopping point as at this step we get the lowest test perplexity of the GRU model. The ROUGE scores calculated using these trained networks are shown in Table 3 and Table 4 for the GRU and MT-GRU models, respectively. A sample summary generated by the MTGRU model is shown in Fig. 5.

Input Text:
The input is the Introduction of this paper.

Generated Summary:
1. Summarization has been the topic explored as a challenge of text semantic understanding
2. Recently , _UNK neural networks have emerged as a success in wide range of practical problems
3. In particular , we need to use a new way to evaluate three important questions into the algorithms
4. We use a concept to define the temporal hierarchy of each sentence in the context of paragraph
5. We demonstrate that our model outperforms a conventional _UNK system and significantly lead to optimize
6. In section # , we evaluate the experimental results on our model and evaluate our results in Section #

Figure 5: An example of the generated summary with MTGRU.

MTGRU Output Summary
Section describes the data collection, models and the experimental results.

Input
The paper is structured as follows: Section 2 describes the related works. Section 3 describes the data collection and processing steps. Section 4 describes the proposed models in detail. In section 5, we report the results of our experiments and show the sample generated summaries. In section 6 we analyze the results of our models.

TF-IDF Extracted Summary
In section 5, we report the results of our experiments and show the sample generated summaries.

Figure 6: An example of the output summary vs the extracted targets

4 Discussion and Future Work

The ROUGE scores obtained for the summarization model using GRU and MTGRU show that the multiple timescale concept improves the performance of the conventional seq2seq model without the presence of highly complex architectural hierarchies. Another major advantage is the increase in training speed by as much as 1 epoch. Moreover, the sample summary shown in Fig. 5 demonstrates that the model has successfully generalized on the difficult task of summarizing a large paragraph into a one line salient summary.

In setting the τ timescale parameters, we follow (Yamashita and Tani, 2008) . We gradually increase τ as we go up the layers such that higher layers have slower context units. Moreover, we experiment with multiple settings of τ and compare the training performance, as shown in Fig. 7. The τ of MTRGU-2 and MTRGU-3 are set as {1, 1.42, 2, 2.5} and {1, 1, 1.25, 1.25}, respectively. MTGRU-1 is the final model adopted in our experiment described in the previous section. MTGRU-2 has comparatively slower context layers and MTGRU-3 has two fast and two slow context layers. As shown in the comparison, the training performance of MTRGU-1 is superior to the remaining two, which justifies our selection of the timescale settings.

The results of our experiment provide evidence that an organizational process akin to functional differentiation occurs in the RNN in language tasks. The MTGRU is able to train faster than the conventional GRU by as much as 1 epoch. We believe that the MTRGU expedites a type of functional differentiation process that is already occurring in the RNN, by explicitly guiding the layers into multiple timescales, where otherwise this temporal hierarchical organization occurs more gradually.
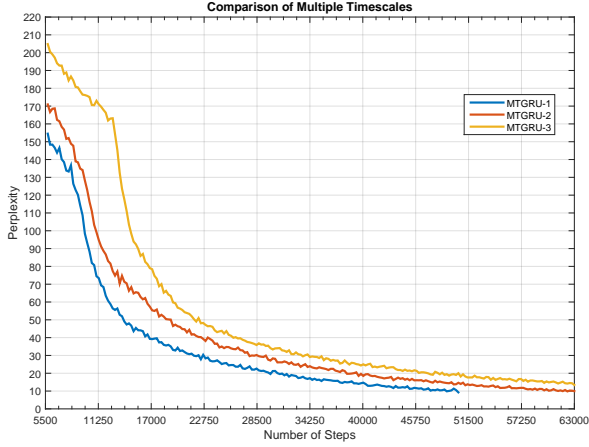
74

Figure 7: Comparison of Training performance between multiple time constants.

In Fig. 6, we show the comparison of a generated summary of the input paragraph to an extracted summary. As seen in the example, our model has successfully extracted the key information from multiple sentences and reproduces it into a single line summary. While the system was trained only on the extractive summary, the abstraction of the entire paragraph is possible because of the generalization capability of our model. The seq2seq objective maximizes the joint probability of the target sequence conditioned on the source sequence. When a summarization model is trained on source-extracted salient sentence target pairs, the objective can be viewed as consisting of two subgoals: One is to correctly perform saliency finding (importance extraction) in order to identify the most salient content, and the other is to generate the precise order of the sentence target. In fact, during training, we observe that the optimization of the first subgoal is achieved before the second subgoal. The second subgoal is fully achieved only when overfitting occurs on the training set. The generalization capability of the model is attributable to the fact that the model is expected to learn multiple points of saliency per given paragraph input (not only a single salient section corresponding to a single sentence) as many training examples are seen. This explains how the results such as those in Fig. 6 can be obtained from this model.

We believe our work has some meaningful implications for seq2seq abstractive summarization going forward. First, our results confirm that it is possible to train an encoder-decoder model to perform saliency identification, without the need to refer to an external corpus at test time. This

has already been shown, implicitly, in previous works such as Rush et al. (2015; Nallapati et al. (2016), but is made explicit in our work due to our choice of data consisting of paragraph-salient sentence pairs. Secondly, our results indicate that probabilistic language models can solve the task of novel word generation in the summarization setting, meeting a key criteria of abstractive summarization. Bengio et al. (2003) originally demonstrated that probabilistic language models can achieve much better generalization over similar words. This is due to the fact that the probability function is a smooth function of the word embedding vectors. Since similar words are trained to have similar embedding vectors, a small change in the features induces a small change in the predicted probability. This makes a strong case for RNN language models as the best available solution for abstractive summarization, where it is necessary to generate novel sentences. For example, in Fig. 5, the first summary shows that our model generates the word "explored" which is not present in the paper. Furthermore, our results suggest that if given abstractive targets, the same model could train a fully abstractive summarization system.

In the future, we hope to explore the organizational effect of the MTGRU in different tasks where temporal hierarchies can arise, as well as investigating ways to effectively optimize the timescale constant. Finally, we will work to move towards a fully abstractive end-to-end summarization system of multi-paragraph text by utilizing a more abstractive target which can potentially be generated with the help of the Abstract from the articles.

5   Conclusion

In this paper, we have demonstrated the capability of the MTGRU in the multi-paragraph text summarization task. Our model fulfills a fundamental requirement of abstractive summarization, deep semantic understanding of text and importance identification. The method draws from a well-researched phenomenon in the human brain and can be implemented without any hierarchical architectural complexity or additional memory requirements during training. Although we show its application to the task of capturing compositional hierarchies in text summarization only, MTGRU also shows the ability to enhance the learning

speed thereby reducing training time significantly. In the future, we hope to extend our work to a fully abstractive end-to-end summarization system of multi-paragraph text.

Acknowledgment

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *journal of machine learning research*, 3(Feb):1137–1155.

Matthew M Botvinick. 2007. Multilevel structure in behaviour and in the brain: a model of fuster's hierarchy. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1485):1615–26, September.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555.

Udo Hahn and Inderjeet Mani. 2000. The challenges of automatic summarization. *Computer*, 33(11):29–36, November.

Stefan Heinrich, Cornelius Weber, and Stefan Wermter, 2012. *Artificial Neural Networks and Machine Learning – ICANN 2012: 22nd International Conference on Artificial Neural Networks, Lausanne, Switzerland, September 11-14, 2012, Proceedings, Part I*, chapter Adaptive Learning of Linguistic Hierarchy in a Multiple Timescale Recurrent Neural Network, pages 555–562. Springer Berlin Heidelberg, Berlin, Heidelberg.

Sepp Hochreiter, Yoshua Bengio, and Paolo Frasconi. 2001. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In J. Kolen and S. Kremer, editors, *Field Guide to Dynamical Recurrent Networks*. IEEE Press.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Karen Sparck Jones. 2007. Automatic summarising: the state of the art. *Information Processing and Management: an International Journal*, 43(6):1449–1481.

Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. *CoRR*, abs/1506.01057.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 71–78. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8.

H. P. Luhn. 1958. The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2(2):159–165, April.

D. Meunier, R. Lambiotte, A. Fornito, K. D. Ersche, and E. T. Bullmore. 2010. Hierarchical modularity in human brain functional networks. *ArXiv e-prints*, April.

Ramesh Nallapati, Bing Xiang, and Bowen Zhou. 2016. Sequence-to-sequence rnns for text summarization. 4th International Conference on Learning Representations - Workshop Track (ICLR 2016).

Ani Nenkova, Sameer Maskey, and Yang Liu. 2011. Automatic summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011*, HLT '11, pages 3:1–3:86, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rainer W. Paine and Jun Tani. 2004. Motor primitive and sequence self-organization in a hierarchical recurrent neural network. *Neural Networks*, 17(89):1291 – 1309. New Developments in Self-Organizing Systems.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389. Association for Computational Linguistics, Lisbon, Portugal.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes,

N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Yuichi Yamashita and Jun Tani. 2008. Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. *PLoS Comput Biol*, 4(11):1–18, 11.