# Improving Neural Abstractive Text Summarization with Prior Knowledge

**Position Paper**

Gaetano Rossiello[1], Pierpaolo Basile[1], Giovanni Semeraro[1], Marco Di Ciano[2], and Gaetano Grasso[2]

[1] Department of Computer Science, University of Bari "Aldo Moro"
{firstname.lastname}@uniba.it
[2] InnovaPuglia S.p.A.
{m.diciano,g.grasso}@innova.puglia.it

**Abstract.** Abstractive text summarization is a complex task whose goal is to generate a concise version of a text without necessarily reusing the sentences from the original source, but still preserving the meaning and the key contents. In this position paper we address this issue by modeling the problem as a sequence to sequence learning and exploiting Recurrent Neural Networks (RNN). Moreover, we discuss the idea of combining RNNs and probabilistic models in a unified way in order to incorporate prior knowledge, such as linguistic features. We believe that this approach can obtain better performance than the state-of-the-art models for generating well-formed summaries.

## 1 Introduction

Information overload is a problem in modern digital society caused by the explosion of the amount of information produced on both the World Wide Web and the enterprise environments. For textual information, this problem is even more significant due to the high cognitive load required for reading and understanding a text. Automatic text summarization tools are thus useful to quickly understand a large amount of information.

The goal of summarization is to produce a shorter version of a source text by preserving the meaning and the key contents of the original. This is a very complex problem since it requires to emulate the cognitive capacity of human beings to generate summaries. For this reason, text summarization poses open challenges in both natural language understanding and generation. Due to the difficulty of this task, research focused on the *extractive* aspect of summarization, where the generated summary is a selection of relevant sentences from the source text in a copy-paste fashion [16] [9]. Over the past years, few works have been proposed to solve the *abstractive* problem of summarization, which aims to produce from scratch a new cohesive text not necessarily present in the original source [17] [16].

Abstractive summarization requires deep understanding and reasoning over the text, determining the explicit or implicit meaning of each element, such as words, phrases, sentences and paragraphs, and making inferences about their properties [14] in order to generate new sentences which compose the summary.

Recently, riding the wave of prominent results of modern deep learning models in many natural language processing tasks [2] [10], several groups have started to exploit deep neural networks for abstractive text summarization [15] [4] [13]. These deep architectures share the idea of casting the summarization task as a neural machine translation problem [1], where the models, trained on a large amount of data, learn the alignments between the input text and the target summary through an attention encoder-decoder paradigm. In detail, in [15] the authors propose a feed-forward neural network based on neural language model [3] with an attention-based encoder, while the models proposed in [4] and [13] use the attention encoder into a sequence-to-sequence framework modeled by RNNs [18]. Once parametric models are trained, a decoder module greedily generates a summary, word by word, through a beam search algorithm.

The aim of these works based on neural networks is to provide a fully data-driven approach to solve the abstractive summarization task, where the models learn automatically the representation of relationships between the words in the input document and those in the output summary without using complex handcrafted linguistic features. Indeed, the experiments highlight significant improvements of these deep architectures compared to extractive and abstractive state-of-the-art methods evaluated on various datasets, including the gold-standard DUC-2004 [12] using various variant of ROUGE metric [11].

## 2 Motivation

The proposed neural attention-based models for abstractive summarization are still in an early stage, thus they show some limitations. Firstly, they require a large amount of training data in order to capture a good representation that properly maps good (soft) alignments between original text and the related summary. Moreover, since these deep models learn the linguistic regularities relying only on statistical co-occurrences of words over the training set, some grammar and semantic errors can occur in the generated summaries. Finally, these models work only at sentence level and are effective for sentence compression rather than document summarization, where both input text and target summary consist of several sentences.

In this position paper we argue about our ongoing research on abstractive text summarization. Taking up the idea of casting the summarization task as a sequence-to-sequence learning problem, we study approaches to infuse prior knowledge into a RNN in a unified manner in order to overtake the aforementioned limits. In the first stage of our research we focus on methodologies to introduce syntactic features, such as part-of-speech tags and named entities.

We believe that informing the neural network about the specific role of each word during the training phase may led to the following advantages: introducing information about the syntactical role of each word, the neural network can tend to learn the right collocation of words by belonging to a certain part-of-speech class. This can improve the model avoiding grammar errors and producing well-formed summaries. Furthermore, the summarization task lacks of availability of data required to train the models, especially in specific domains. The introduction of prior knowledge can help to reduce the amount of data needed in the training phase.

## 3 Methodology

In this section we provide a general view of our proposed model starting from a formal definition of the abstractive summarization problem to a discussion of the proposed approach aimed at introducing a prior knowledge into neural networks.

### 3.1 Model

Let us denote by $x = \{x_1, x_2, \ldots, x_n\}$ and $y = \{y_1, y_2, \ldots, y_m\}$ with $n > m$, two sequences, where $x_i, y_j \in V$ and $V$ is the vocabulary. $x$ and $y$ represent sequences of words of the input text and the output summary over the vocabulary $V$, respectively.

The summarization problem consists in finding an output sequence $y$ that maximizes the conditional probability of $y$ given an input sequence $x$:

$$\arg\max_{y \in V} P(y|x) \tag{1}$$

The conditional probability distribution $P$ can be modeled by a neural network, with the aim of learning a set of parameters $\theta$ from a training set $T = \{(x^1, y^1), \ldots, (x^k, y^k)\}$ of source text and target summary pairs. Thus, the problem is to find the right parameters that represent a good approximation of probability $P(x|y) = P(x|y;\theta)$.

The parametric model is trained to generate the next word in the summary, conditioned by previous words and the source text. Then, the conditional probability $P$ can be factored as follows:

$$P(y|x;\theta) = \prod_{t=1}^{|y|} P(y_t|\{y_1, \ldots, y_{t-1}\}, x; \theta) \tag{2}$$

Since this is a typically sequence to sequence learning problem, the parametric function that computes the conditional probability can be modeled by RNNs using a encoder-decoder paradigm. Figure 1 shows a graphical example. The encoder is a RNN that reads one token at time from the input source and returns a fixed-size vector representing the input text. The decoder is another RNN that generates words for the summary and it is conditioned by the vector representation returned by the first network.

Formally,

$$P(y_t|\{y_1, \ldots, y_{t-1}\}, x; \theta) = g_\theta(h_t, c) \tag{3}$$

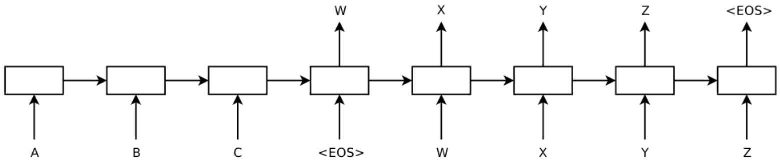

**Fig. 1.** An example of encoder-decoder paradigm for sequence to sequence learning [18].

At the time $t$ the decoder RNN computes the probability of the word $y_t$ given the last hidden state $h_t$ and the context input $c$, where

$$h_t = g_\theta(y_{t-1}, h_{t-1}, c) \tag{4}$$

The vector context $c$ is the output of the encoder and encodes the representation of the whole input source. This vector is fundamental to inform the decoder about the input representation during the generation of the next word. Some attention-based mechanisms [15] [1] [4] are integrated to help the network to remember certain aspects about the input. The good performance of the whole architecture often depends on how these attention-based components are modeled.

A simpler way to model $g_\theta$ is using an Elman RNN [7]. Hence:

$$h_t = sigmoid(W_1 y_{t-1} + W_2 h_{t-1} + W_3 c) \tag{5}$$

$$P(y_t | \{y_1, \ldots, y_{t-1}\}, x; \theta) = softmax(W_4 h_t + W_5 c) \tag{6}$$

where $W_i$ are matrices of parameters learned during the training phase.

In tasks involving language modeling, variants of RNNs have shown impressive performance and they solve the vanishing gradient problem. These variants are Long-Short Term Memory (LSTM) [8] and Gated Recurrent Unit (GRU) [5].

Finally, the decoder generates summaries by assigning probability values word by word. In order to find a sequence that maximize the equation (1), a beam search algorithm is commonly used.

The whole architecture is inspired by [18] and [1], which use this setting to solve a machine translation problem learning soft alignments between source and target sentences. However, the summarization problem has two significant differences. The words in both sequences $x$ and $y$ share the same vocabulary $V$ and the problem is constrained by the length of the input source, which must be shorter than the target summary. Despite in [15], [4] and [13] the authors adopt the same paradigm to solve the abstractive summarization task by taking in account these constraints, their proposals regard only summarization of unique sentences. This constraint makes the summarization closer to a machine translation problem, where the length of the source and the target are similar. Conversely, for a document level of summarization, where the summary is far more shorter than the original text, the length constraint is stronger. Designing neural models to solve summarization at document o multi-document level is a promising future direction that we want to explore.

### 3.2 Proposed approach

In our preliminary research we focus on techniques to incorporate prior knowledge into a neural network. We start by taking into account only lexical and syntactic information, such as part-of-speech and named entities tags. The core idea is to replace the softmax of each RNN layer with a log-linear model or a probabilistic graphical model, like factor graphs. This replacement does not arise any problem because the softmax function converts the output of the network into probability values, where the softmax can be seen as a special case of the extended version of RNN [6]. Thus, the use of probabilistic models allows to condition the probability value, given an extra feature vector that represents the lexical and syntactic information of each word.

We believe that this approach can learn a better representation of the input context vector during the training and it can help the decoder in the generation phase. In this way, the decoder can assign to the next word a probability value which is related to the specific lexical role of that word in the generated summary. This can allow the model to decrease the number of grammar errors in the summary, even using a smaller training set since the linguistic regularities are supported by the extra vector of syntactic features.

## 4 Evaluation Plan

We plan to evaluate our models on gold-standard datasets for the summarization task, such as DUC-2004 [12], Gigaword [15] and CNN/DailyMail [13] corpus, as well as on a local government dataset of documents made available by InnovaPuglia S.p.A. (consisting of projects and funding proposals) using several variants of ROUGE [11] metric.

ROUGE is a recall-based metric which assesses how many n-grams in generated summaries appear in the human reference summaries. This metric is designed to evaluate extractive methods rather than abstractive ones, thus the former would be advantaged.

The evaluation in summarization is a complex problem and it is still an open challenge for three main reasons. First, given an input text, there are different summaries that preserve the original meaning. Furthermore, the words that compose the summary could not appear at all in the original source. Finally, ROUGE metric cannot measure the quality of grammar structure of the generated summary. To overcome these issues we plan an in-vivo experiment with a user study.

## 5 Conclusions and Future Work

In this position paper we outlined our ongoing research on abstractive text summarization using deep learning models. The abstractive summarization is a harder task than extractive summarization, where the techniques produce a summary by selecting the most relevant sentences from an input source text. We propose a novel approach to combine probabilistic models with neural networks in a unified way in order to incorporate prior knowledge such as linguistic features. Using this approach, as future work we plan to integrate also semantic knowledge so that the neural network can be able to learn jointly word and knowledge embeddings by exploiting knowledge bases and lexical thesaurus. Moreover, the generation of abstractive summaries from documents or multiple documents is another promising direction that we want to investigate.

## References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR abs/1409.0473 (2014)
2. Bengio, I.G.Y., Courville, A.: Deep learning (2016), book in preparation for MIT Press
3. Bengio, Y., Ducharme, R., Vincent, P., Janvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. 3, 1137–1155 (2003)
4. Chopra, S., Auli, M., Rush, A.M., Harvard, S.: Abstractive sentence summarization with attentive recurrent neural networks. (2016), http://harvardnlp.github.io/papers/naacl16_summary.pdf
5. Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555 (2014)
6. Dymetman, M., Xiao, C.: Log-linear rnns: Towards recurrent neural networks with flexible prior knowledge. CoRR abs/1607.02467 (2016)
7. Elman, J.L.: Finding structure in time. Cognitive Science 14(2), 179–211 (1990)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. 9(8), 1735–1780 (1997)
9. Jones, K.S.: Automatic summarising: The state of the art. Information Processing & Management 43(6), 1449–1481 (2007)
10. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature 521, 436–444 (2015)
11. Lin, C.Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Proc. of the ACL-04 Workshop. p. 10. Association for Computational Linguistics (2004)
12. Litkowski, K.C.: Summarization experiments in duc. In: Proc. of DUC 2004 (2004)
13. Nallapati, R., Xiang, B., Zhou, B.: Sequence-to-sequence RNNs for text summarization. CoRR abs/1602.06023 (2016)
14. Norvig, P.: Inference in text understanding. In: AAAI. pp. 561–565 (1987)
15. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. In: Proc. of EMNLP 2015, Lisbon, Portugal. pp. 379–389 (2015)
16. Saggion, H., Poibeau, T.: Automatic text summarization: Past, present and future. In: Multi-source, Multilingual Information Extraction and Summarization, pp. 3–21. Springer (2013)
17. Salim, N.: A review on abstractive summarization methods. Journal of Theoretical and Applied Information Technology 59(1), 64–72 (2014)
18. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Proc of NIPS. pp. 3104–3112 (2014)