Hierarchical Summarization with Auxiliary Information using Recursive Feed-Forward Neural Network

Kailash Karthik Saravana Kumar[1] and Dr. N. R. Subramanian[2]

[1, 2] Department of Computer Science and Engineering, National Institute of Technology - Tiruchirappalli

Abstract

The explosion in availability of online content has spurred intense research in the area of automatic text summarization. Contemporary research in extractive summarization focus on deep learning techniques and feature engineering. They rely only on the textual body of the document to be summarized for sentence extraction. In contrast to this, we propose a scalable and fully data-driven technique that leverages a simple feed-forward network recursively, making use of auxiliary information that occur in the context of newswire documents like image captions and document title. The single document summarization framework proposed is evaluated on the CNN dataset and is shown to perform comparable to the state-of-the-art techniques. It is shown that the proposed methodology outperforms other modern techniques that summarizes without any auxiliary information based on the information content of the generated summaries. A metric to evaluate summarization systems called Pragmatic Performance Index (PPI) is proposed and used in summarization performance evaluation.

Keywords

Text Summarization, News Summarization, Neural Networks, Natural Language Processing.

1 INTRODUCTION

Text summarization refers to that task of distilling the most important information from a text and to produce an abridged version for a task and user (Mani and Maybury, 1999). According to Radev et al. (2002), a summary is no longer than half the original text(s) and is usually significantly less than that. With the ubiquity of the internet and the massive generation of online data, there is a growing demand for automated summarization in applications like search engines, market reviews, medical and bio-medical document consolidation. The global digitization of news data has opened avenues to identify existing and emerging trends, mentions of people/organizations and timeline-based evolution of storylines (Liepins et al., 2017). Text summarization as a technique is of importance within the natural language processing community as it is a vital component in many linguistic tasks like information retrieval and knowledge extraction.

Text summarization techniques can be classified based on multiple criteria (Nenkova and McKeown, 2012; Saggion and Poibeau, 2013): method of summary text generation – extractive and abstractive; purpose of the summary generated – generic and query/topic/user-focused; and purview of summary – single and multi-document. While extractive techniques aim to generate extracts or verbatim subsections of the original document as summaries (Cheng and Lapata 2016; Nallapati, Zhai and Zhou 2017), abstractive techniques aim to produce abstracts that summarize the concepts from the source using verbal innovation (See, Liu and Manning 2017). Such methods require a deep understanding of language and meaning and is an area of active research, though abstractive systems still are outperformed by most of the recent extractive techniques.

There are three independent tasks that are comprised in text summarization: 1. Construction of an intermediate representation, focusing on relevant aspects and features of the source text; 2. Information structuring and ordering by sentence scores based on the intermediate representation; 3. Sentence realization by selection of several relevant sentences (Mehdi Allahyari et al. 2017). This paper addresses single-document generic extractive summarization and explores enhanced methods of sentence realization.

Research on automatic text summarization began in the late 1950s with the seminal work of Luhn (1958), extracting word and phrase frequency as features for sentence selection. Later techniques used position in the document (Baxendale, 1958) and key phrases (Edmundson, 1969) as sentence features to determine their relevance in the summary. Recent techniques have modelled the summarization task as a classification problem that is solved using machine learning techniques (Bazrfkan et al. 2014; Sarkar et al. 2011; Kim Y 2014). Naive Bayes, Support Vector Machines (SVM), Hidden Markov Models (HMM) and Conditional Random Fields (CRF) are among the most common machine learning techniques used for summarization.

The two problems with contemporary methods that the proposed method seeks to solve are: 1. the focus on only the main body of the document for summarization

and 2. the inherent challenges with deep learning. The proposed method makes use of auxiliary information surrounding the text like image captions to determine features for sentence relevance. Deep learning methods are replaced with a scalable technique that leverages a simple feed forward network that recursively feeds on input documents of various sizes.

Extractive summarization techniques generally focus on the text body of documents for sentence extraction. Though traditional methods have often relied on manually defined features for sentence extraction, recent machine learning methods (deep learning in particular) have negated the necessity for human-engineered features. Early feature generation techniques used parts-of-speech information and keywords (Mani 2001), sentence length and position within a document (Radev et al. 2014), content word and document frequency adjusted based on context (Nenkova et al. 2006). Contemporary methods use machine learning models to map sentences to a vector space using pre-trained language models (Yin and Pei 2015). With the emergence of recurrent neural networks, techniques to handle documents of dissimilar lengths have been devised (Cheng and Lapata 2016) that report state of the art performance.

The challenge in summarization using only the main body of documents is that it requires a deep understanding of language to extract good cues for sentence relevance. But documents, newswire articles in particular, often have many types of auxiliary information such as image captions, tabular data and title that either directly or indirectly capture the essence of the document. Previous literature has explored the use of these auxiliary information in the generation of sentence features as early as the late 1960s (Edmundson 1969). Figure 1 shows an example document taken from a CNN newswire that illustrates the same (Narayan et al. 2017). As observed, the golden summary focuses on topics that are mentioned as part of the image captions like “Japanese businesses test using robot in stores”, thereby validating the premise that auxiliary information surrounding the document can give significant cues for sentence extraction.

Though deep learning can be traced back to the late 90s (Yann LeCun 1998), it was with the ImageNet success (Hinton et al. 2012) that the use of deep learning techniques proliferated in the field of natural language processing (T Du and Vijay Shankar 2013). Though most contemporary research is based on such methods, deep learning presents several complex challenges that impediment its widespread usage as a de facto technique in language processing (Angelov and Sperduti 2016).

The major challenges with deep learning are the requirement for a vast amount of training data, over-fitting with respect to training data, hyper-parameter optimization and the requirement of high-performance hardware for the training of models (Sze et al. 2017). Among these, the necessitation of large volumes of digitized training data prevents its use in many applications. For example, deep learning may not produce adequate results when applied to summarization of research manuscripts in a niche field in which only a handful of papers have been published. Neither will it be useful for region specific applications in countries like India where document digitization is still an ongoing process. The requirement for high performance GPUs/TPUs for training make it impossible for widespread usage by researchers who don’t have access to cutting-edge hardware. Most importantly, the “Blackbox” nature of deep learning models make it unusable in domains where verification of the decision-making process is required.

Japan Companies Experiment with Work Robots

Tokyo (CNN) - A crowd gathers near the entrance of Tokyo's upscale Mitsukoshi Department Store, which traces its roots to a kimono shop in the late 17th century.

Fitting with the store's history, the new greeter wears a traditional Japanese kimono while delivering information to the growing crowd, whose expressions vary from amusement to bewilderment.

It's hard to imagine the store's founders in the late 1600's could have imagined this kind of employee. (…)

- Japanese businesses test using robots in stores
- Can this robot read your emotions?
- Japan's robot revolution

- Toshiba tests robotic greeter at upscale Tokyo department store
- More Japanese businesses are testing out robots as possible solution to Japan's shrinking workforce

Figure 1 A CNN newswire article. The figure shows (in vertical order) the headline, news body (truncated), image captions as auxiliary information and the highlights of the article that are treated as the golden summary.

In this paper, we explore the advantages of using auxiliary information, while using a hierarchical network architecture featuring the recursive application of a simple feed-forward neural network. We evaluate our models using ROUGE metrics on the CNN dataset (Herman et al. 2015). A new metric to evaluate the performance of automatic summarization systems that accounts for their practical usability is introduced. Experimental results show that the proposed technique’s performance is comparable to the state-of-the-art methods that use deep learning strategies. We also demonstrate the positive impact of auxiliary information on quality of the generated summaries.

2 PROBLEM FORMULATION

In this section, we present a formal definition of the extractive summarization. Given a document D with a sequence of sentences $D = \{s_1, s_2, s_3 \dots s_n\}$ and a sequence of auxiliary information $A = \{a_1, a_2, a_3 \dots a_p\}$, the objective

of the summarization task is to produce an extractive summary *S* by selecting m (< n) sentences {s_i, s_j ... s'_m} from the document D. The summarization process involves labelling each sentence s_i in D with a label $l_i \in \{0, 1\}$, indicating its presence or absence in the summary generated. The summary to be generated is assumed to have a predefined length *k*. Thus, the problem objective can also be stated as obtaining an optimal '*k*' sentence subset of the document as the extracted summary.

This problem is modelled as a supervised learning problem whose objective is to maximize the likelihood of the label sequence L = { l_i : if $s_i \in S \rightarrow l_i = 1$; else $l_i = 0$} given the input document D and model parameters Θ.

$$\log p(L|D;\Theta) = \mathbf{\Sigma} \log p(l_i|D;\Theta)$$

The next section presents our proposed technique to generate extractive summaries using auxiliary information through the recursive application of a simple feed-forward neural network.

3 HIERARCHICAL SUMMARIZATION

The hierarchical summarizer consists is based on a feed-forward neural network of the simplest kind – consisting of a single input, hidden and output layer each. While the document is fed as a fixed-length vector to the input layer, computations performed by the hidden layer result in the production of summary labels from the output layer. The two main aspects of using this architecture include the engineering of the document vector and the recursive application of the network to handle documents of varying length.

Machine learning models are essentially probabilistic and thus work with numbers to train and predict. Hence sentence embedding techniques are employed to convert the sequence of words in the input document into a vector fixed length that can be fed to the network. The embedding strategy used consists of skip-gram based model that takes into consideration sub-word character n-grams (Bojanowski et al. 2017). Each character n-gram is associated with a vector and words are represented as a summation of these sub-word units.

The embedding technique requires a language model to generate word and sentence vectors. A language model estimates the probability distribution of various linguistic units like words, phrases and sentences. Though pre-trained models like the wiki-english model are available for use, a custom model is trained on a domain specific corpus for the summarization task. The language model training data is a large dataset of newswire article texts. The generated language model is used to train a skip-gram based embedding model.

The word embedding model thus generated is then used to generate vectors for each sentence in the document. There are many contemporary methods to generate sentence vectors including averaging word vectors, average of word vectors augmented with TF-IDF, paragraph vectors (Mikolov 2014) and skip-thought vectors (Kiros, Ryan, et al. 2015). The proposed system

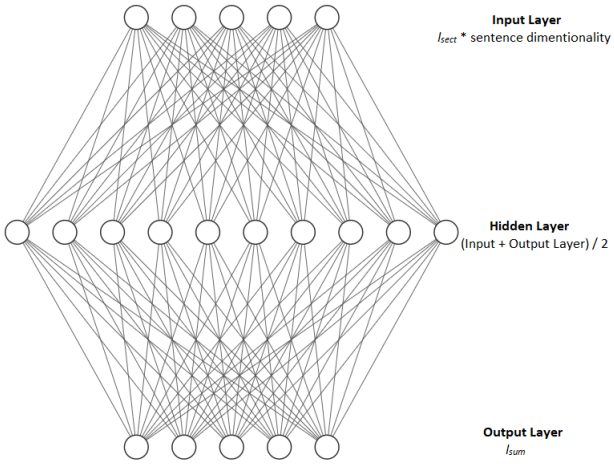

Figure 2 Proposed Neural Network Architecture

generates sentence vectors as the mean of normalized word vectors (Joulin et al. 2016).

The fixed-length sentence vectors generated consider only the main body content of the document. Thus, the vectors are augmented with similarity measures to each auxiliary information. In the proposed system, the auxiliary information considered are the title of the document and image captions that occur with the news document. Each sentence vector is augmented with additional features that bear a truth or false value depending on the sentence's similarity with an auxiliary information sentence. The mean value of the previously generated sentence vectors is used as the reference values to be set as the truth value for these additional features while false values are set as zero. Cosine distance of the fixed-length sentence vectors is used to determine similarity of two sentences. The sentence bearing the highest similarity value to an auxiliary content sentence is given a truth value for that additional feature while all other sentences are given a value of zero for that feature.

The content of documents exhibit variation in length with respect to two aspects – sentence length and sentence count. Though sentence length has been normalized through the generation of sentence vectors of uniform dimensionality, the variation in the number of sentences needs to be handled. Contemporary methods employ recurrent neural networks (RNNs) and encoder decoder networks to solve the problem of varying input length. But these methods are computationally expensive and require massive amounts of training data to produce adequate results and models that are not over-fitted. An alternative approach to handle this variation is leveraged in the proposed technique that recursively summarizes fragments of the document (Sinha, Yadav and Gahlot 2018).

Each document of length l_{doc} is fragmented into sections, each comprising of a fixed number of sentences l_{sect}. The last section is padded with empty sentences if required. This fragmentation results in the generation of ceil (l_{doc} / l_{sect}) fixed-size sections. A section vector of length (l_{sect} * dimensionality of sentence vector) is generated by

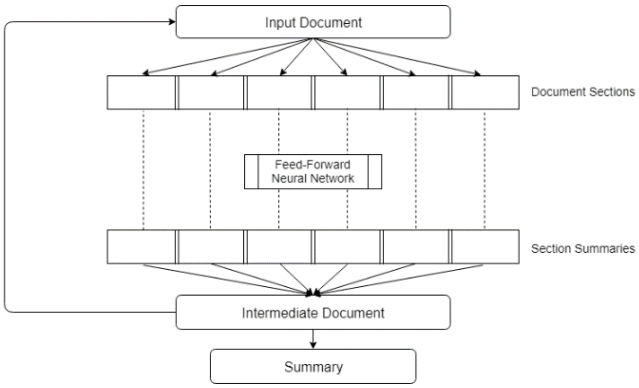

Figure 3 Architecture of Hierarchical Summarization System

appending all the individual sentence vectors. The vectors generated are of uniform length and are used to feed the network. The section length is a hyper-parameter that is optimized based on the validation set performance.

ReLU activation (Nair and Hinton 2010) is applied at the hidden layer while a softmax activation function is used to generate the network output. The output generated represents the probability of the corresponding input sentence being extracted as part of the summary. The error metric used for network training is categorical cross-entropy and the network is further enhanced with the Adam optimizer. The proposed neural network architecture is presented in Figure 2. Since the network uses a single hidden layer neural network, it is computationally inexpensive and has a quick training latency due to the presence of fewer trainable network parameters as compared to deep neural networks.

The generation of the extractive summary begins with the partitioning of the entire document into smaller sections. The target summary length l_{sum} (< $l_{sect,}$ the section length) is pre-determined and is a parameter to the proposed technique. Each document section is fed into the network to generate the summary distribution across its sentences. The sentences are ranked in order of their probabilities and the top l_{sum} sentences are extracted as the summary of the section. The extracted section summary sentences are then concatenated to create an intermediate document. This generated document is then partitioned into intermediate sections and fed to the same network. This process is recursively performed until the length intermediate document satisfies the condition $l_{int\_doc} < l_{sum}$. In other words, the network is recursively used to summarize the sections of the intermediate document until the output summary length falls short of the target summary length. When the summary length criterion is satisfied the intermediate document is asserted as the summary of the original document. Figure 3 presents the overall architecture of the proposed system.

Thus, the recursive application of a simple feed-forward neural network on document sections can generate a summary for the entire document consisting of the l_{sum} sentences that are most relevant in terms of information content.

A new performance evaluation metric "*Pragmatic Performance Index (PPI)*" is proposed in this paper. It is a measure of the performance that can be extracted from a summarization model for any practical application. It not only takes into consideration the accuracy of the model being evaluated, but also the computational applicability of the model in a real-time scenario. The motive behind the proposal of this metric is that as the complexity of modern systems increases, its scope for usage in a wider range of applications reduces due to lack of data, hardware and training time. Thus, the proposed metric makes a trade-off between high semantic performance and low system complexity. *PPI* is defined as follows:

PPI = Harmonic Mean ($Perf_{Absolute}$, $Perf_{Relative}$)

where

$Perf_{Absolute}$ is the absolute performance of the system under evaluation as measured by metrics like precision, recall or ROUGE,

$Perf_{Relative}$ is the relative performance of the system under evaluation measured as absolute performance per computational unit.

The proposed metric scales the absolute performance of a summarization system with its computational complexity. Since harmonic mean is involved in the mathematical definition of the index, to score a high PPI a system would have to have high values for both absolute and relative performance. In the above definition of relative performance, a computational unit is expounded and will depend on the application. A computational unit can be a unit of training time or a unit of processing capacity (a gigabyte of GPU memory for example).

4 EXPERIMENTAL SETUP

This section presents the experimental setup for the evaluation of the proposed system. The training and test dataset are discussed. The evaluation strategy for the performance of the proposed network architecture is also presented. A brief description of the baseline comparison model is also provided.

4.1 Dataset – Training and Test Data

The proposed model was trained on the extended version of the CNN dataset (Hermann et al. 2015; Narayan et al. 2017). Documents annotated with sentence relevance labels were required for the training of the proposed model. Each sentence was labelled as 1 if present or 0 if absent in the golden summary.

The CNN dataset has evolved with time since its first collection by Svore, Vanderwende and Burges in 2007. It was subsequently enhanced by Woodsend and Lapata in 2010 and Herman et al. in 2015, transforming into a corpus that sets a benchmark for summarization techniques. CNN articles come with "story highlights" that are treated in literature as the golden summary.

In the experiments conducted, the dataset was annotated in the Nallapati, Zhai and Zhou. 2017 style. The basic CNN dataset is augmented to include auxiliary

information like image captions and article title. Each article is associated with a single title whereas the multiple image captions can be associated with each article. The number of image captions varies from 0 to 414 per article, with 3 being the mean captions-per-article count.

The network is trained on the entity anonymised version of the dataset using the standard splits of Herman et al. (2015) for training, validation and testing (90,545, 1,220 and 1,093 documents respectively). The evaluation of the system for the above dataset is based on two variants of ROUGE evaluation – ROUGE-1 and ROUGE-2.

4.2 Evaluation – Comparison Techniques

The proposed system was evaluated against multiple contemporary extractive summarization techniques. The baseline for comparison is the simple extraction of the first three sentences of each document as its summary. This is referred to as LEAD-3 in the rest if this paper.

The proposed system was also compared to the extraction system of Narayan et al (2017). This system is referred to as SIDENET as it leverages side information for summarization. It leverages an encoder-decoder architecture assembled by recurrent and convolutional neural networks.

The second comparison system was the extractive summarization system proposed by Sinha, Yadav and Gahlot. This system doesn't make use of any auxiliary information but relies on a recursive architecture as opposed to deep learning techniques for summarization. This system is referred to as RECURNET in this paper.

The final comparison system was a Cosine-based-ILP technique proposed by Kristian Woodsend and Mirella Lapata. This technique leverages PCFG and dependency trees to generate a phrase-based representation of the document. The model then learns to combine phrases as summaries using Integer Linear Programming strategies. This system is referred to as COS-ILP in this paper.

4.3 Implementation

The training data was first used to train word embeddings using a skip-gram based character N-gram model with a context window size 5, embedding vector dimension of 300, negative sample size 5 and the skip-gram negative sampling (SGNS) loss function. Pre-trained word embeddings from the wiki-english model were used to aid the supervised learning process. Using the generated embedding model, sentence vectors were constructed for each document's tokenized sentence and its associated auxiliary information.

The auxiliary information sentences were compared to the main body using the cosine similarity measure and the auxiliary features were set to truth values for the most similar sentences for each auxiliary sentence. This leads to sentence embeddings having a dimensionality of 302 in the proposed model.

The neural network model was implemented using TensorFlow (Abadi et al. 2015). The documents were trained by batching into groups of 10, running batches for 10 epochs each. The model was trained using the Adam optimizer (Kingma and Ba 2015) having a learning rate of 0.001. The loss was evaluated by means of the categorical cross-entropy metric.

The network was trained for various values of the hyper-parameters, namely the section length l_{sect}, its performance being evaluated on the validation set.

5 RESULTS AND DISCUSSIONS

This section details the performance evaluation of the proposed model as compared to other contemporary extractive summarization strategies. For automatic evaluation of the quality of the summaries generated, a recall-oriented metric ROUGE (Lin and Hovy 2003) is used to compare generated summaries against the gold standard. The information content of the summaries is determined by comparing their ROUGE-1 (*R1*) and ROUGE-2 (*R2*) scores. ROUGE-*n* captures the extent of overlap of *n*-grams between the generated and expected summaries.

MODEL	R1	R2
LEAD	49.0	19.1
SIDENET	53.9	20.8
RECURNET	52.1	20.0
COS-ILP	51.4	19.7
Proposed System	53.6	20.6

Table 1 Relative performance of the compared systems with respect to story highlights. The proposed system has a ROUGE score that matches the state-of-the-art SideNet

MODEL	R1	R2
SIDENET	52.7	20.2
Proposed System	52.1	19.9

Table 2 Relative performance of the proposed system and SideNet with respect to manual summaries.

Results are reported on full length summaries, each matching the number of sentences in the gold standard. The LEAD summary is taken as the first three sentences guided by the average length of the gold summaries in the corpus (Narayan et al. 2017). The first experiment aimed to determine the effect of the hyper-parameter section length on the summary generated. The section length was

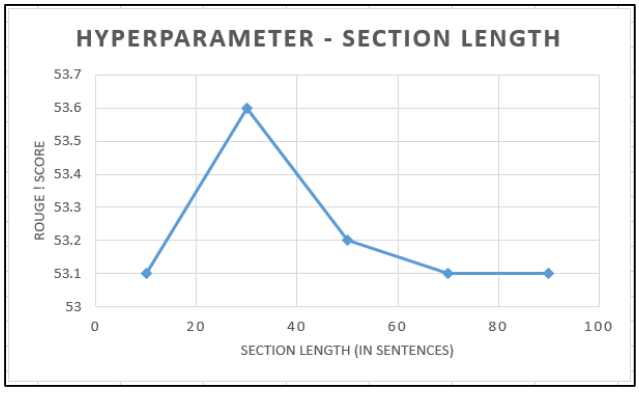

Figure 4 The effect of section length as a hyper-parameter in the proposed model

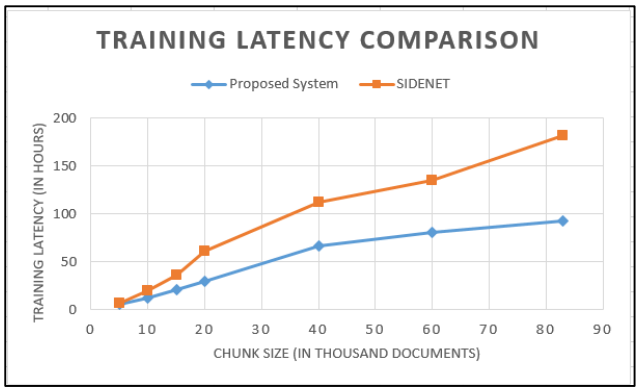

Figure 5 A comparison of the training latency of the proposed model and RECURNET for various document chunk sizes of the corpus.

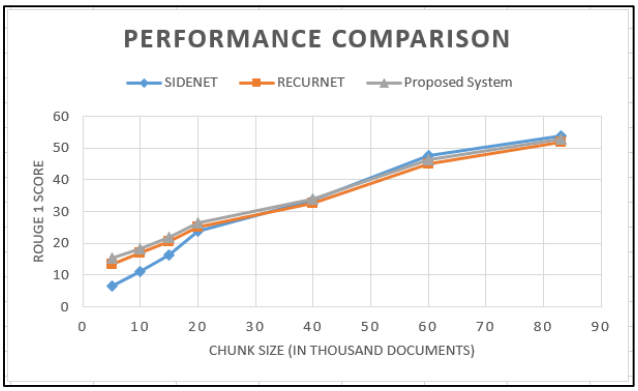

Figure 6 A comparison of the summarization performance for the systems in comparison for various document chunk sizes using the ROUGE 1 metric.

varied from 10 to 90 at intervals of 20 sentences each. As shown in Figure 4, we observed that the proposed system performed the best with a section length of 30 sentences. Both R1 and R2 metrics validated similar trends across variation in section length, peaking with a ROUGE-1 score of 53.6 and ROUGE-2 score of 20.6 for the 30-sentence section length.

The summaries of the systems compared are evaluated on the test set and the results are presented in Table 1. The proposed model is shown to have comparable performance to SIDENET, which employs complex deep learning strategies using an encoder-decoder network. Though the system doesn't make use of deep neural networks, it is able to generate good summaries due to its recursive architecture.

The system performs considerably better than the other contemporary techniques of LEAD, COS-ILP and RECURNET. The proposed system can better identify salient sentences in the summary due to the presence of additional context in the form of auxiliary information. The impact of this additional information in the automatic generation of high-quality summaries is thus evident.

A second experiment was conducted on a sample set of 25 randomly chosen documents from the corpus. Summaries were manually generated by a group of 5 human summarizers in agreement, all proficient speakers of the English language. These summaries were treated as the gold standard as the performance evaluation was repeated on all the compared systems. The results of this evaluation are presented in Table 2. We see similar results as the previous analysis with the proposed methodology having a similar performance to the state-of-the-art SIDENET.

To experimentally prove the advantages of the proposed method, an experiment was conducted to measure the computational requirement of the models in comparison. The proposed system and SIDENET neural networks were trained on the CNN corpus. The training was done chunking the corpus into subsets of various sizes and the training time was noted for all subsets for a given chunk size. The documents that remain after forming subsets of the chunk size are omitted from the analysis. This implies that the number of subsets for each chunk size is $Floor(Size_{Corpus}/Size_{Chunk})$. The training time for a given chunk size was determined as the mean training time for all subsets of that chunk size to avoid any subset-specific factors that may influence the training latency. The models were trained on an NVidia GTX 950M GPU using the GPU variant of TensorFlow. As shown in Figure 5, it was observed that the proposed system trains at a latency that is a fraction of what is required by SIDENET. To be able to provide near-state-of-the-art performance at a fractional cost (latency and complexity) is the first advantage of the proposed model.

A subsequent experiment was conducted to validate the premise that the proposed model would be able to generalize better even with a smaller training dataset. For each chunk size as mentioned above, all corresponding subsets were used to train the model independently and the trained models were evaluated using the test data. The ROUGE score for a particular chunk size is the mean score of all its corresponding subsets. As shown in Figure 6, it is observed that the proposed system consistently outperforms RECURNET for all chunk sizes. The proposed system also outperforms SIDENET for smaller chunk sizes while its performance remains comparable at larger chunk sizes.

The ratio of performance to computational requirement (α training time of the model) is observed as shown in Figure 7. It is thus evident that the proposed

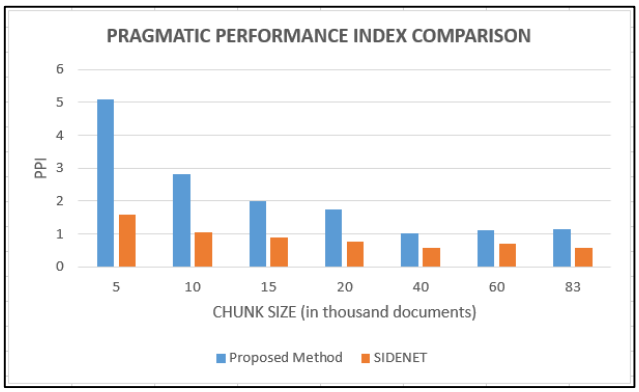

Figure 7 A comparison of the Pragmatic Performance Indices (PPIs) of the systems in comparison for various document chunk sizes.

system can both achieve a comparable absolute performance to SIDENET while outperforming it significantly based on performance relative to computational complexity. It outperforms RECURNET both in absolute and relative performance consistently.

Further investigation was done on the systems in comparison by measuring their Pragmatic Performance Index (PPI). For the conducted analysis, a unit of training latency was regarded as a unit of computational complexity. It was observed that for all chunk sizes and the corpus in its entirety, the proposed system achieved a higher PPI as compared to the other systems (Table 4). It can be inferred that the proposed model has a higher performance with regards to practical applications of text summarization.

6 CONCLUSION

In this paper, a recursive neural network architecture for extractive text summarization was presented. The technique proposed leverages auxiliary information like image captions and document header as additional context. The system was evaluated on the CNN newswire dataset against the story highlights and manually generated summaries as gold standard. The experiments conducted show the impact of auxiliary information in the extraction of salient sentences as part of the generated summary. It is also shown that the proposed method, though computationally inexpensive as compared to deep learning strategies, achieves comparable performance.

Future work may include the exploration of alternative recursive strategies on the neural network architecture. Moreover, other kinds of auxiliary information like social media comments, search query logs and news sub-category can be leveraged to further improve the summaries generated. This hierarchical summarization technique can also be applied to various other domains like scientific manuscript summarization to observe its cross-domain performance.

REFERENCES

Aakash Sinha, Abhishek Yadav. 2018. "*Extractive Text Summarization using Neural Networks.*"

Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mane, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viegas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; and Zheng, X. 2015. "*TensorFlow: Large-scale machine learning on heterogeneous systems.*"

Angelov, Plamen and Alessandro Sperduti. 2016. "*Challenges in Deep Learning*."

Bazrfkan, Mehrnoosh and M. Radmanesh. 2014. "*Using Machine Learning Methods to Summarize Persian Texts.*"

Cheng, J., and Lapata, M. 2016. "*Neural summarization by extracting sentences and words.*" In Proceedings of ACL, 484–494.

Dragomir R Radev, Eduard Hovy, and Kathleen McKeown. 2002. "*Introduction to the special issue on summarization.*" Computational linguistics 28, 4 (2002), 399–408.

Du, Tianchuan, and V. Shanker. 2009. "*Deep Learning for Natural Language Processing.*" Eecis. Udel. Edu: 1-7.

Edmundson, H. P. 1969. "*New methods in automatic extracting.*" Journal of the ACM 16(2):264–285.

Hermann, K. M.; Kocisk y, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; and Blunsom, P. 2015. "*Teaching machines to read and comprehend.*" In NIPS 28, 1693–1701.

Inderjeet Mani. 1999. "*Advances in Automatic Text Summarization.*" Mark T. Maybury (Ed.). MIT Press, Cambridge, MA, USA.

Kim, Y. 2014. "*Convolutional neural networks for sentence classification.*" In Proceedings of EMNLP, 1746–1751.

Kingma, D. P., and Ba, J. 2015. "*Adam: A method for stochastic optimization.*" In Proceedings of ICLR.

Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Torralba, A., Urtasun, R. & Fidler, S. 2015. "*Skip-Thought Vectors.*" CoRR, abs/1506.06726.

LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. 1998. "*Gradient-Based Learning Applied to Document Recognition.*" Proceedings of the IEEE (p./pp. 2278--2324).

Liepins, R.; Germann, U.; Barzdins, G.; Birch, A.; Renals, S.; Weber, S.; van der Kreeft, P.; Bourlard, H.; Prieto, J. a.; Klejch, O.; Bell, P.; Lazaridis, A.; Mendes, A.; Riedel, S.; Almeida, M. S. C.; Balage, P.; Cohen, S. B.; Dwojak, T.; Garner, P. N.; Giefer, A.; Junczys-Dowmunt, M.; Imran, H.; Nogueira, D.; Ali, A.; Miranda, S. a.; Popescu-Belis, A.; Miculicich Werlen, L.; Papasarantopoulos, N.; Obamuyide, A.; Jones, C.; Dalvi, F.; Vlachos, A.; Wang, Y.; Tong, S.; Sennrich, R.; Pappas, N.; Narayan, S.; Damonte, M.; Durrani, N.; Khurana, S.; Abdelali, A.; Sajjad, H.; Vogel, S.; Sheppey, D.; Hernon, C.; and Mitchell, J. 2017. "*The summa platform prototype.*" In Proceedings of EACL: Software Demonstrations, 116–119.

Lin, C.-Y., and Hovy, E. 2003. "*Automatic evaluation of summaries using n-gram co-occurrence statistics.*" In Proceedings of NAACL, 71–78.

Mani, I. 2001. "*Automatic Summarization.*" Natural language processing. John Benjamins Publishing Company.

Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. 2017. "*Text Summarization Techniques: A Brief Survey.*" In Proceedings of arXiv, USA, July 2017, 9 pages.

Mikolov, Tomas, et al. 2013. "*Distributed representations of words and phrases and their compositionality.*" Advances in neural information processing systems.

Nallapati, R.; Zhai, F.; and Zhou, B. 2017. "*Summarunner: A recurrent neural network based sequence model for extractive summarization of documents.*" In Proceedings of AAAI, 3075–3081.

Nenkova, A.; Vanderwende, L.; and McKeown, K. 2006. "*A compositional context sensitive multi-document summarizer: Exploring the factors that influence summarization.*" In Proceedings of ACM SIGIR, 573–580.

P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, "*Enriching Word Vectors with Subword Information.*"

P. B. Baxendale. 1958. "*Machine-made index for technical literature: an experiment.*" IBM J. Res. Dev. 2, 4 (October 1958), 354-361.

Radev, D.; Allison, T.; Blair-Goldensohn, S.; Blitzer, J.; C¸ elebi, A.; Dimitrov, S.; Drabek, E.; Hakim, A.; Lam, W.; Liu, D.; Otterbacher, J.; Qi, H.; Saggion, H.; Teufel, S.; Topper, M.; Winkel, A.; and Zhang, Z. 2004. "*MEAD — A plat- form for multidocument multilingual text summarization.*" In Proceedings of LREC, 699–702.

Saggion, H., & Poibeau, T. 2013. "*Automatic Text Summarization: Past, Present and Future. Multi-source, Multilingual Information Extraction and Summarization.*"

Sarkar, Kamal, Mita Nasipuri, and Suranjan Ghose. 2011. "*Using machine learning for medical document summarization.*" International Journal of Database Theory and Application 4.1: 31-48.

Svore, K. M.; Vanderwende, L.; and Burges, C. J. C. 2007. "*Enhancing single-document summarization by combining ranknet and third-party sources.*" In Proceedings of EMNLPCoNLL, 448–457.

Vinod Nair and Geoffrey E. Hinton. 2010. "*Rectified linear units improve restricted boltzmann machines.*" In Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10), Johannes Fürnkranz and Thorsten Joachims (Eds.). Omnipress, USA, 807-814.

Vivienne Sze, Yu-Hsin Chen, Joel Emer, Amr Suleiman. 2016. "*Hardware for Machine Learning: Challenges and Opportunities.*"

Woodsend, K., and Lapata, M. 2010. "*Automatic generation of story highlights.*" In Proceedings of ACL, 565–574.

Yin, W., and Pei, Y. 2015. "*Optimizing sentence modeling and selection for document summarization.*" In Proceedings of IJCAI, 1383–1389.