Extractive Summarization of Individual News Articles

Using Structured Prediction Energy Networks

Gaurav Bhaskar Gite

Advisor: Kathleen McKeown

School of Engineering and Applied Science
Columbia University

This dissertation is submitted for the degree of
Masters

Columbia University June 2017

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Gaurav Bhaskar Gite
June 2017

Acknowledgements

I would like to acknowledge the guidance and advice of my thesis advisor Prof.Kathleen McKeown and my thesis committee members Prof. Julia Hirschberg and Prof. Smaranda Muresan. Special thanks to Christopher Kedzie, who has been helping me from the beginning of this research. I would also like to thank my friends at the NLP lab Noura Farra, Elsbeth Turcan and others.

Abstract

We introduce two novel architectures based on Structured Prediction Energy Networks (SPENs) for extractive summarization of news articles. SPENs is a framework for structured learning and can be effectively applied to multi-label classification problems. It has two major components, a *feature network* which produces feature representation of the inputs and an *energy network* which captures the relationship between the input representation and output labels.

Extractive summarization of single documents can be framed as a multi-label classification problem, where every sentence is classified as included or excluded in the summary. In both the proposed architectures, the feature network involves a Convolutional Neural Network (CNN) to learn sentence representation. In the first architecture, the energy network tries to identify salient sentences based on the context of the sentence and its position. The second proposed architecture's energy network attempts to form clusters of sentences based on the topics discussed in the article. It aims to diversify the summary and reduce repetitiveness by selecting sentences from different topic clusters.

We also extend the New York Times Annotated Corpus by adding hundreds of thousands of recent article-summary pairs.

The baseline for the task of single document summarization on news articles is obtained by taking the first few sentences of the article. This baseline is noted to be very strong due to the journalistic convention of putting important information at the start of the news article. Experimental results show that the sentence salience based model performs significantly better than the baseline. The topic clustering based model lags slightly behind the baseline on larger length summaries but outperforms the baseline on the shorter length summaries. Currently, our summarizer is used as part of another project where we are sending weekly feedbacks to the residents in one of the Columbia-owned building about their electricity consumption. We hypothesize that news summaries about environmental issues and concerns could persuade the residents to reduce their electricity consumption.

Table of contents

1 Introduction **1**

1.1 Single Document Summarization 1

1.2 Structured Prediction Energy Networks 2

1.3 Introduction to Proposed Models 4

1.4 Related work 6

2 Datasets **7**

2.1 Dailymail 7

2.2 New York Times 8

3 Model Architecture **10**

3.1 Feature Network 10

3.2 Energy Network 12

3.2.1 Sentence Salience based Energy Equation 12

3.2.2 Topic Cluster based Energy Equation 13

3.3 Training 13

4 Results and Discussion **15**

4.1 Dailymail 15

4.2 New York Times 16

4.3 Document Understanding Conference 2002 18

5 Future Work and Conclusion **19**

5.1 Future Work 19

5.2 Conclusion 19

References **21**

Chapter 1

Introduction

1.1 Single Document Summarization

Single document summarization is an important task in information retrieval and natural language processing. Summarization is often divided into two types, extractive summarization and abstractive summarization. Extractive summarization methods on a single document produce a summary by selecting several sentences from the document and concatenating them. Abstractive summarization techniques produce a summary by concisely paraphrasing the information present in the document. Extractive summarization methods generally produce grammatically correct summaries, something that is difficult to achieve in abstractive summarization systems. But the extractive summaries can be incoherent and can have a longer length.

Various methods for sentence extraction have been proposed. Most of these methods rely on sentence-level features like sentence position, length of the sentence, word frequency and presence of proper nouns. Some of the previously used methods of extractive summarization are based on binary classification (Kupiec et al., 1995), graph based models (Erkan and Radev, 2004) and integer linear programming (Woodsend and Lapata, 2010).

Recently, there has been a surge of interest in the application of neural networks for different natural language processing tasks including summarization. This is primarily due to the introduction of new large scale corpora. Recurrent Neural Networks (RNN) have produced state-of-the-art results in tasks like machine translation (Sutskever et al., 2014) and dependency parsing (Kiperwasser and Goldberg, 2016). Rush et al. in 2015 applied a Recurrent Neural Network (RNN) for abstractive summarization of the first sentence of a news article to generate the title. Cheng and Lapata, 2016 and Nallapati et al., 2017 proposed Neural Networks based models for the extractive summarization of news articles. Their models was trained on the Dailymail corpus (Hermann et al., 2015) thousands of

news articles and had promising experimental results. Our proposed models also use neural networks to learn effective sentence representations and perform sentence level extractive summarization. It is trained on the Dailymail corpus and our newly introduced New York Times article-summary pairs. The training and testing is performed separately on each of the two corpora, Dailymail and New York Times.

The baseline for the task of single document summarization on news articles is obtained by selecting the leading sentences of the article. Due to the journalistic convention of putting important information at the start of the news article this baseline has been noted to be very strong. The generic single document summarization task was discontinued by Document Understanding Conference (DUC) after the first two years because no automatic summarization system could outperform the simple baseline. Nenkova, 2005 noted that human performance was significantly higher than the baseline on the DUC datasets, showing that while the baseline was quite strong, there was still room for improved automatic summarization systems. As large scale corpora for the task of summarization are now available along with encouraging results from extractive neural network models, interest in single document summarization is increasing.

1.2 Structured Prediction Energy Networks

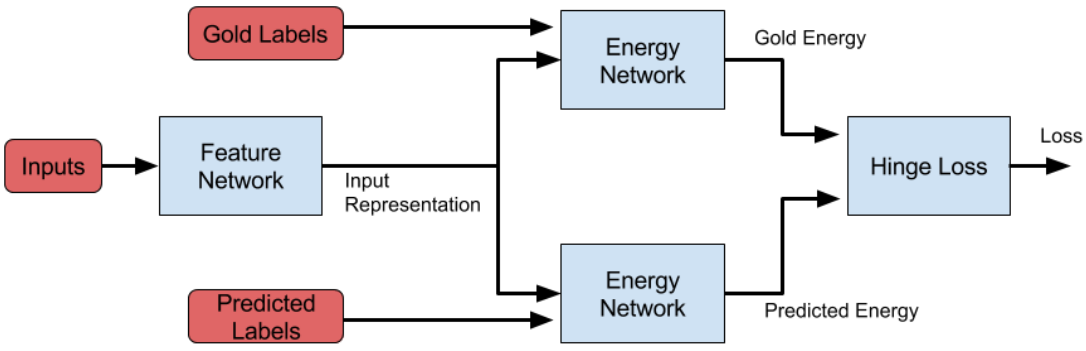


Fig. 1.1 Basic architecture of SPENs

Belanger and McCallum, 2016 introduced a flexible framework for structure learning tasks called Structured Prediction Energy Networks (SPENs). SPENs can be effectively applied to a multi-label classification problem, where an input x is mapped to a binary vector y. The basic architecture of SPENs consists of a feature network and an energy network. The feature network tries to produce the feature representations for the inputs. The energy network is a parameterized function that captures the correlation between the

feature representations and the output labels; more likely output label sequence are assigned a lower energy by the model. During model training, the parameters for the feature and energy network are adjusted to assign the lowest energy states to the input feature/output labels pairs observed in the training set.

In mathematical terms, given an input x and gold output labels y, the feature network gives the input feature representation $F(x)$ where F is some form of neural network. The energy network gets $F(x)$ and y as inputs and returns an energy value $E(F(x),y)$. E is a parameterized energy function. The objective of the network is to obtain minimum energy for the gold output labels, $y = \arg\min_{y'} E(F(x),y')$. In order to ensure that the energies of the gold output labels are well separated from erroneous configurations, we train our model to minimize the hinge loss between the gold labels and the model's current lowest energy predicted labels. The training for the model works as follows:

1. Determine the gold energy $E(F(x),y)$ from input feature representation $F(x)$ and gold output labels y.

2. Predict the output labels $\bar{y}$ which gets the least energy value. It is given by $\bar{y} = \arg\min_{y'} E(F(x),y')$. The energy value corresponding to the predicted labels is called predicted energy, $E(F(x),\bar{y})$.

3. Calculate the hinge loss as $max(0, E(F(x),y) - E(F(x),\bar{y}) + \Delta(y,\bar{y}))$ where Δ is the error function (example - squared loss) between gold output labels and predicted output labels.

4. Minimize the hinge loss by performing gradient descent w.r.t. to the feature and energy network parameters.

The value of the hinge loss is proportional to the difference of the gold energy and predicted energy. By reducing the hinge loss, the network attempts to decrease the difference between the gold energy and predicted energy. It tries to achieve the objective of gold output labels having the least energy value. $\Delta(y,\bar{y})$ is the squared loss between the predicted labels and the gold labels and helps in reducing the difference between them. Predicting the output labels (step 2) with minimum energy has an exponential complexity in the length of labels. It is approximately and efficiently calculated by finding minima for the energy equation $E(F(x),y')$ in the region where each label value is between [0,1] and then rounding this label value. While searching for minima, $F(x)$ is assumed to be constant and gradient descent is performed iteratively on the output labels y'. Each output label is bounded within the region [0,1] for every instance of the iteration. During an iteration, if a label exceeds the boundary

limit of 0 or 1, it is projected back into the search space. In the inference phase, the predicted output labels are the output of the system.

One of the major benefits of the SPENs is that by modeling the energy equation a practitioner can utilize domain knowledge and guide the model to learn discriminative features for the structured output. Another advantage is that during prediction time, sequence labellings are predicted jointly, rather than in a greedy fashion typically employed in RNN models. Joint prediction can be helpful in sequence labeling tasks by avoiding common pitfalls like label bias. Additionally, joint prediction can capture complex long range sequence interactions in a way that greedy prediction in RNN's cannot.

1.3 Introduction to Proposed Models

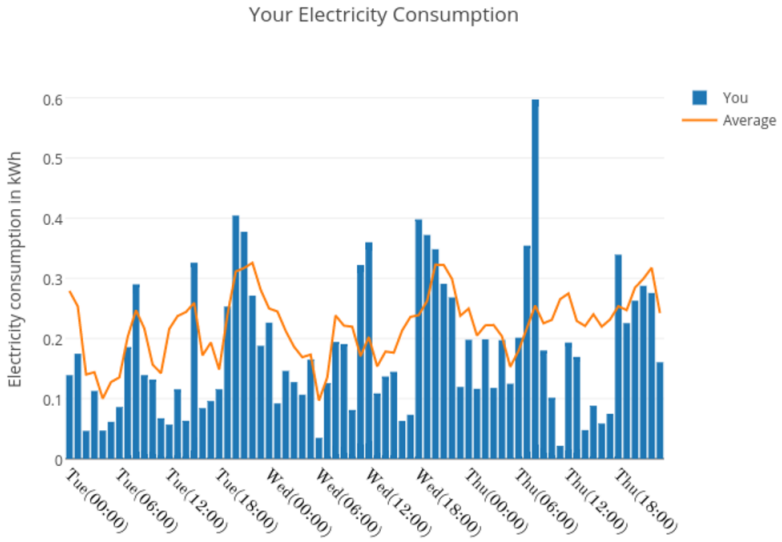
Single document summarization of a news article can be framed as a multi-label classification problem, where every sentence in the article can be labeled as included or excluded. We have developed two SPENs based architecture which can be used to extract sentences from a document for the summary. In both the proposed models, a Convolutional Neural Network (CNN) is used as the feature network. CNN based sentence encoders were introduced in Kim, 2014 for sentimental classification and were also used in Cheng and Lapata, 2016. We also tried using a Recurrent Neural Network (RNN) but found that CNN worked slightly better and was significantly faster to train.

The first proposed model's energy function is based on the idea of selecting salient sentences. It seeks to build a filter at each sentence position to identify an important sentence. The second proposed model's energy function is based on the notion of diversifying and broadening the coverage of the summary. The model attempts to form sentence level clusters of topics presented in the article and then select sentences from different clusters, consequently making the summary more diverse, reducing the repetitiveness and improving the coverage.

We have trained separate models with the same architecture for the Dailymail and New York Times corpus. The evaluation was also performed independently on the test samples of each dataset. ROUGE metrics (Lin, 2004) and F-1 were used to evaluate the performance. We also evaluated our system on the Document Understanding Conference (DUC) 2002 dataset. Experimental results show that the sentence salience based model outperforms the topic cluster based model. On the Dailymail corpus, the sentence salience based model significantly outperforms the baseline of leading sentences. The topic cluster based model does better on short length summaries but does not beat the baseline on longer length in terms

of ROUGE scores. We believe that the topic cluster based model doesn't do well because of the lower variation in topics discussed in a news article.

We are using our topic cluster based summarizer system in another project which aims at reducing the electricity consumption of residents in the buildings by giving them personalized feedback on their electricity usage. We hypothesize that including a news article summary about environmental issues and concerns could motivate the residents to reduce their usage. A sample feedback message is shown in the figure below. Currently, we are sending feedback twice a week, every Monday and Friday, to an experimental group of around 90 apartments in a Columbia-owned building. The experimental group is compared to a control group of around 100 apartments from the same building. This system uses MCMC Gibbs Sampling inference to learn the Multivariate Gaussian Mixture Model, to form groups of apartments based on the response to the feedback. In each group, we use logistic regression to identify the important aspects of the feedback message such as sentiment of the message, presence of graph, information about the other similar apartments, etc. to personalize the feedback.



This feedback cycle (Monday-Thursday), you used electricity that caused 4.34 kg of greenhouse gas emissions. That's on a daily basis 25.27% less than your previous feedback cycle. Great job!

For the month, we estimate you would be spending $22.24 on electricity.

We think this news article may interest you:
A warming world will drive famine, drought and natural disasters, creating the sort of societal and economic upheaval that strengthens terrorists' recruiting efforts, a new report warns. Read more at http://nypost.com/2017/04/20/climate-change-could-fuel-the-global-rise-of-terrorism/

Hope this feedback message helps.

Fig. 1.2 Sample feedback message.

Overall the contributions of this work are:

- Introducing two novel architectures based on SPENs which capture different aspects of extractive summarization.
- Adding hundreds of thousands of recent articles along with their summaries to the New York Times Annotated Corpus[1].
- Applying our model in another project involving summarization of environmental news articles.

1.4 Related work

We are aware of two models which are applied to the task of extractive summarization using neural networks, Cheng and Lapata, 2016 and Nallapati et al., 2017. Cheng and Lapata, had applied an encoder-decoder based model for the task of extractive summarization. In their model they have a hierarchical encoder, which first uses CNN based extractors to encode the sentences to sentence embeddings and RNN based extractors to further encode these sentence embeddings to a document vector. They used an attention-based decoder mechanism to classify each sentence sequentially. Nallapati et al. have presented an approach where they encode the sentences using a RNN and then use a bidirectional GRU cell based RNN to alter the sentence embeddings based on the neighboring sentence embeddings. Their work came out after we had started this research. A document embedding is obtained by averaging the sentence embeddings. They give scores to each sentence based on the position of the sentence, content of the sentence, salience with respect to the document and redundancy with respect to the summary generated so far. Based on this score they make a prediction if the sentence is to be included or excluded. Both these models produce labels in a sequential manner. Our sentence salience model is somewhat closer to the Nallapati et al. but we only look at the content and position of the sentence and completely ignore the salience with respect to the document or redundancy in the summaries. We also use a different framework which is more powerful because it can capture interactions within the labels which is difficult to capture in the encoder-decoder framework. Also, our model produce label for all the sentence positions at once.

Chapter 2

Datasets

2.1 Dailymail

The dailymail corpus was introduced by Hermann et al., 2015 for the question-answer task on news articles. It was modified by Cheng and Lapata, 2016 and Nallapati et al., 2016 to obtain the article-summary pairs for summarization. It contains 196,557 training examples, 12,147 validation examples and 10,396 test examples. The summary in the article-summary pair is a human-written highlight of the article and can be considered as the gold abstractive summary. An example of an article and gold abstractive summary is given below. For the task of extractive summarization, we need to split the article into sentences and give labels (0 - excluded, 1 - included) to every sentence. The NLTK based tokenizer was employed to tokenize the article into sentences. In order to give a label, sentences are picked so that the ROUGE-2 (Lin, 2004) score is maximized between the selected sentences and the gold abstractive summary (highlights). These selected sentences are labeled as included, forming the gold extractive summary, while the rest are labeled as excluded. As determining the best combination of sentences which maximizes the ROUGE-2 score is a problem which is exponential in the length of the article, a greedy strategy discussed in Nallapati et al., 2016 was employed where one sentence is added at a time to the set of selected sentences. The greedy strategy loops over every sentence which is not included and selects a sentence such that there is a maximum improvement in the ROUGE-2 score for the selected sentences. It stops adding sentences when the number of sentences selected is equal to the number of sentences in the highlight or no sentence is found which improves the ROUGE-2 score. For faster training of our models we only consider the top 10 sentences of the article, hence each article is truncated to a length of 10 sentences. It was found that on average 3-4 sentences were selected by the greedy algorithm from the top 10 sentences. We created the sentence-label pairs for each article on the training dataset, the validation dataset and the test dataset.

Note that during the inference, we compare the system summary to the gold abstractive summary to measure the performance of the system. The below example also shows the gold extractive summary selected for the article.

Table 2.1 Example from Dailymail Corpus - Article Sentences, Highlights and Gold Extractive Summary.

Article Sentences	Move over Grumpy Cat - the latest sourpuss to become an internet sensation is scowling feline Pompous Albert. The grey cat with tufty hair and a permanent frown, from Salt Lake City, has already racked up a remarkable 44.1K followers on Instagram. Albert, who's a Selkirk Rex cat, was also named after his famous lookalike - Albert Einstein, thanks to his wild locks, which are a trademark of his breed. Scroll down for video Albert is from Salt Lake City and is a constant source of entertainment to his 44.1k Instagram followers Albert is named after famous physician Albert Einstein, due to sharing the same white-grey curly hair The Instagram account shows images of Albert in a various situations, frowning and glaring at the camera - his feline brow furrowed disapprovingly. The hit social media account is run by his owners Mike and Susan Singleton in Salt Lake City, USA, who've penned his bio, which reads: 'Rejected show cat, but I'll show them.' Each picture of the fluffy cat is accompanied by a sarcastic caption, giving the animal a sinister and funny internal dialogue. One snap, in which he stares fiercely into the camera reads: 'I'm not grumpy or angry, I'm just smarter and better looking than you.' Another is captioned: 'Albert's Office Tip: This is the look you give a boss who wants you to work over the weekend.'
Highlight	1. Albert is breed of cat called Selkirk Rex known for its wild, tufty fur 2. He is named after Albert Einstein thanks to his untamed grey coat 3. His owners live in Salt Lake City and regularly post pictures of their cat
Gold extractive summary	Albert, who's a Selkirk Rex cat, was also named after his famous lookalike - Albert Einstein, thanks to his wild locks, which are a trademark of his breed. Albert is from Salt Lake City and is a constant source of entertainment to his 44.1k Instagram followers The hit social media account is run by his owners Mike and Susan Singleton in Salt Lake City, USA, who've penned his bio, which reads: 'Rejected show cat, but I'll show them.'

2.2 New York Times

The New York Times Annotated Corpus[1] contains New York Times articles along with their summaries from 1987 to 2007. We make an extension to it by adding recent news articles for the years 2007 to 2016. The New York Times API allows developers to get their news article meta-data which includes human-written summaries for the article, authors, keywords, and article URL. The API doesn't give the article text so we scraped it from the URL for every

article which had an abstract for the years 2007 to 2016. But unlike the Dailymail highlights which have a broader coverage of the article, most abstracts in the NYT were shorter in length. These abstracts were written to entice the users to click on the article link - this aligns with our other project's goal where we want the resident to click on the environmental news article. We have ignored the articles with an abstract length of fewer than 200 characters. In order to keep the size of the dataset comparable to that of the Dailymail Corpus, we are only using articles from the last 11 years (2006-2016) to train and test our model. This helps in faster training time. More data would have required longer training time. At the end for the current task of extractive summarization, we had a total of 198,016 training articles, 10,520 validation samples, and 10,520 testing samples. After the article-summary pairs were obtained we conducted a similar exercise as discussed in the Dailymail section. We split the article into sentences using the NLTK tokenizer and label the sentences using the greedy ROUGE algorithm, to maximize the ROUGE-2 score between the selected sentences and human-written summaries. We found that on average 2-3 sentences were selected by the greedy algorithm for this corpus. Again, we are only considering the top 10 sentences of the article. An example of the article, human-written abstracts (gold abstractive summary) and gold extractive summary are shown in table 2.2.

Table 2.2 Example from NYT Corpus - Article Sentences, Highlight, Gold Extractive Summary.

Article Sentences	A home health aide to I. M. Pei, the renowned 98-year-old architect, has been charged with assaulting him inside his home in New York, the authorities said. Mr. Pei told the police that the aide, Eter Nikolaishvili, 28, grabbed his right forearm and forcefully twisted it on Dec. 13. The authorities said Mr. Pei's arm was bruised and bleeding after the attack. The police investigated for two weeks before arresting Ms. Nikolaishvili on Tuesday. She was arraigned in Manhattan Criminal Court on a charge of felony assault and was released without bail. The aide's lawyer did not immediately return a phone call seeking comment. Mr. Pei's designs include the John F. Kennedy Library and Museum in Boston and the glass and steel pyramid at the Louvre in Paris. In 1983, Mr. Pei was awarded the Pritzker Prize, known as the Nobel Prize of architecture.
Highlight	Police say Eter Nikolaishvili, home health aide to renowned architect I M Pei, has been charged with assaulting him inside his Manhattan home; Pei had reported that Nikolaishvili grabbed his right forearm and forcefully twisted it.
Gold extractive summary	A home health aide to I. M. Pei, the renowned 98-year-old architect, has been charged with assaulting him inside his home in New York, the authorities said. Mr. Pei told the police that the aide, Eter Nikolaishvili, 28, grabbed his right forearm and forcefully twisted it on Dec. 13.

Chapter 3

Model Architecture

In this section, we describe our Structured Prediction Energy Networks (SPENs) based models. We have proposed two different models, *Sentence Salience* based model and *Topic Cluster* based model, which intend to learn different aspects of extractive summarization. On a high-level, both models follow the basic architecture of the SPENs, as discussed in section 1.2. Moreover, both models have a similar feature network architecture. The training algorithm for the models is also exactly similar to the previous discussion. But the energy equations are different and hence the model parameters are influenced accordingly. We first explain the feature network which is shared by both models and then the energy equation for the two models are discussed.

3.1 Feature Network

Convolutional Neural Network (CNN) based sentence representation extractors were used by Kim, 2014 for the sentiment classification task. In our models, we have also used a CNN based sentence representation extractor.

Consider a sentence consisting of words $w_1, w_2, \ldots, w_n$. Each word is represented as a parameterized word embedding vector with dimension d. The sentence matrix (S) is a dense column matrix with each row as the word embedding vector for the words in the sentence, $S \in \mathbb{R}^{nxd}$. Feature filters are given by $F_1, F_2, \ldots, F_m$, each with a bias $b_1, b_2, \ldots, b_m$. The dimensionality of the filters is $F_i \in \mathbb{R}^{h_i xd}$, where h_i is the height for the i^{th} filter. In the below diagram there are two filters with height 2 (orange) and 3 (green) respectively for a sentence of 10 words with the word embedding dimension as 5.

For every filter, a convolution operation is performed between the sentence matrix S and the filter F_i. In other words, the filter is centered at every row of S and an element-wise

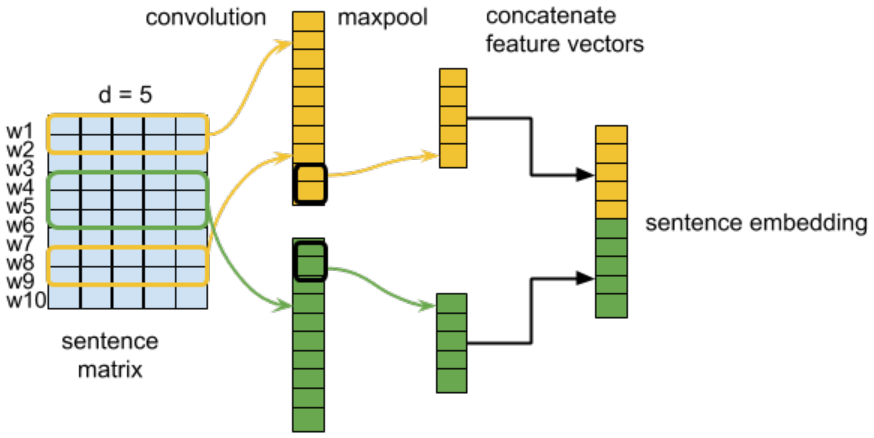


Fig. 3.1 Convolutional Neural Network (CNN) based Sentence Representation Extractor

multiplication is performed between the overlapping parts. This can be expressed as

$$SF_{ij} = relu(S_{j-\frac{h_i}{2}:j+\frac{h_i}{2}-1} \star F_i + b_i) \qquad \forall j \in [1,n]$$

where $\star$ is a element-wise multiplication operation between the slice of the sentence matrix S within the rows $[j-\frac{h_i}{2} : j+\frac{h_i}{2}-1]$ and the filter F_i. The sentence matrix is assumed to be padded with zeros so a filter can overflow when centered around the start and end positions of the sentence. SF_{ij} is the j^{th} entry in the feature vector output (size : n) for filter F_i. Likewise, feature vectors are obtained for other filters. These vectors are max pooled and concatenated to form the sentence embedding. The size of the sentence embedding depends on the number of words in the sentence, the number of filters and the max pool window width. This operation is conducted on every input sentence of the article. The filters and the word embeddings are shared across the sentences. For an input article, a list of all the sentence embeddings is the output of the feature network. The list of sentence embeddings along with output labels constitutes the input to the energy network.

In our experiments, we used 300-dimensional pre-trained embeddings [2] to initialize the word embeddings. These embeddings were trained on word2vec (Mikolov et al., 2013) skip-gram model on Google's 1-billion word benchmark. Words which did not have a pre-trained embedding were initialized randomly. We only considered the words which occurred more than 50 times in the training data. This resulted in a vocabulary size of around 40,000 words. All the other less frequent words were given a common UNK token. Two filters of sizes 1, 2, 3, 4, 5, 6, 7 each were used. The max pool window width was set to 3. We also fixed our sentence length to 50 words. Sentences greater than length 50 were truncated and sentences shorter than length 50 were padded with zeros. This resulted in the sentence

embedding size of 224. We took only the first 10 sentences of the article. Articles shorter than length 10 were padded with zeros. Hence, the output of the feature network was a list of ten 224-dimensional sentence representations. We tried experimenting with the Recurrent Neural Networks (RNN) as feature networks and found that their performance was slightly lower than the CNN. Furthermore, RNN took a lot more time to train than CNN as the GPUs are optimized for the CNN operations.

3.2 Energy Network

SPENs have better opportunities to incorporate domain knowledge by modeling the energy equation, which captures the interactions between the input representation and output labels. Below we discuss the energy equations for the two models.

3.2.1 Sentence Salience based Energy Equation

Energy Equation for the sentence salience based model is given as follows:

$$E(F(x), y) = \sum_{i=1}^{k} \left(y_i s_i W_i - (1 - y_i) s_i W_i \right)$$

$F(x)$ - Input feature representation, given by a list of sentence embeddings obtained from the feature network $[s_1, s_2, \ldots, s_k]$
y - Output labels, where y_i is the label for the i^{th} sentence.
k - Number of sentences in the input. This is 10 in our experiments.
W_i - Weight vector at i^{th} sentence position. Dimension is same as s_i.

For every sentence, the dot product between the sentence embedding s_i and weight vector for the i^{th} position W_i produces a sub-score which is added or subtracted to the energy value. The objective of the model is to have a minimum energy value for the gold output labels. Loss of the network is dependent on the difference of gold energy and predicted energy where predicted energy is the minimum energy value which can be obtained for the predicted labels. During training, the network tries to reduce the loss and by extension the energy function. This forces the model to learn weight vectors, convolutional filter and word embedding parameters such that the energy of the gold label sequence is minimized. The feature network shares the filters and word embedding parameters across the sentences and hence, these parameters would capture the context of the sentence, independent of the sentence position.

However, W_i acts as a filter for the i^{th} sentence position. As a result, the salience of the sentence would be based on the sentence position and the context of the sentence.

3.2.2 Topic Cluster based Energy Equation

Energy equation for the topic cluster based model is given below.

$$E(F(x),y) = -\Bigg(\sum_{i=1}^{k}\Big(\sum_{j=1}^{k}(1-cos(s_i,s_j))y_iy_j + cos(s_i,s_j)y_i(1-y_j)\Big)\Bigg)$$

$F(x)$ - Input feature representation, given by a list of sentence embeddings obtained from the feature network $[s_1, s_2, \ldots, s_k]$
y - Output labels, where y_i is the label for i^{th} sentence.
k - Number of sentences in the input. This is 10 in our experiments.
$cos(s_i, s_j)$ - Cosine distance between s_i and s_j.

Consider the first term in the energy equation, which we refer to as the *summary diversity* term. This term is active when sentence i and j are both included in a summary, i.e. $y_i = y_j = 1$. When this term is active, the energy is minimized if the feature network representations s_i and s_j are pointing in the opposite direction from each other. When predicting a label sequence y, the model will tend to assign more diverse summaries with a lower energy score. The second term of the energy equation, the *summary coverage* term, is active only when $y_i = 1$ and $y_j = 0$. For lower energy value the cosine distance between the included sentence and the excluded sentence must be high. This is achieved when the excluded sentence embedding s_j and the included sentence embedding s_i are aligned. As a result of the coverage term, the excluded sentences would be clustered around the included sentences. We assume that the sentences which are included in the gold extractive summary address different topics and hence, these clusters are based on the topics presented in the article. During training of the model, we hope that the feature network parameters - filters and word embeddings - would be tuned such that the sentence representations obtained from the feature network form clusters of sentences.

3.3 Training

Training for both the models is similar to the algorithm discussed in the introduction of SPENs. For every batch of input articles, sentence representations are obtained using the CNN. Predicted labels corresponding to the minimum energy value are searched by

performing gradient descent on the energy equation assuming the sentence representations as constant. Gradient descent is performed only in the region between 0 and 1 for every label. At each iteration, if a label value moves out of that region it is projected back to the nearest boundary label. At the end of the gradient descent, we round off the labels. The energy value corresponding to the rounded-off predicted labels is the predicted energy. Hinge loss is calculated by taking the summation of the difference of gold energy and predicted energy and the squared loss between the predicted labels and gold labels. The objective is to minimize the hinge loss. We have used a gradient descent based optimizer with decaying learning rate and gradient clipping to perform this optimization.

The initial learning rate for the gradient descent algorithm used while calculating the predicted labels is 0.5. The stopping condition for it is to have squared difference of less than 0.001 between the consecutive predictions or maximum iteration of 50. For the gradient descent on the hinge loss optimization, 0.005 was used as the learning rate with a decay of 75 percentage with every 6 epochs. The maximum gradient norm is set to 5. The training took about 2 days for every model. During prediction, we use predicted labels with the minimum energy as the output labels. It was observed that the network choose 3-4 sentences for the summary.

Chapter 4

Results and Discussion

Following Cheng and Lapata, 2016 we report performance of our models using limited length ROUGE recall at 75 bytes and 275 bytes on the test samples of Dailymail and New York Times separately. We also test our system on the Document Understanding Conference (DUC) 2002 dataset with a limited length of 75 words summary following the official guidelines. ROUGE-1 and ROUGE-2 are reported to indicate the informativeness of the summary and Rouge-L for the fluency. ROUGE scores are calculated between the extractive summary and the gold abstractive summary (human-written highlights or abstracts) of the article. We also report the F-1 scores for the models calculated between the predicted labels and gold extractive labels. Note that ROUGE-2 score is used to produce the gold extractive summary, in other words, labels to the sentences as mentioned in the section 2. But that procedure is independent and does not affect the limited length ROUGE scores between the system summary and the gold abstractive summary.

4.1 Dailymail

Table 4.1 summarizes the results of our models with respect to the baseline of lead-3 sentences and 3 randomly selected sentences. It can be seen that lead-3 sentences is a considerably higher baseline than random baseline. As expected, our learned models have a higher F-1 than the lead-3 baseline. The topic cluster based model performs better than lead 3 on the shorter length summaries at 75 bytes. The sentence salient based model is significantly better than the leading 3 sentences of the articles at both 75 bytes and 275 bytes. F-1 scores also suggest that sentence salient based model performs better than the topic cluster based model.

On a short length of 75 bytes, our models outperform the Cheng and Lapata, 2016 but lag behind on the longer length summaries of 275 bytes length as observed by the rouge scores.

Recently published work by Nallapati et al., 2017 is the state of the art on this dataset and outperform our models.

One of the major difference between our model and other models is that our model predicts the output labels for all the sentences at once. Other models perform a sequential selection of sentences and hence are able to reduce the repetitiveness in the summaries by looking at the previously selected sentences. Our topic cluster based model tries to address repetitiveness but it has to learn that in the model intrinsically. Our sentence salience model doesn't care about other selected sentences at all while making the prediction for a sentence. This is one of our hypotheses for the lower performance of our model compared to other models on longer summary lengths. In the below table, best results of our models are underlined and * indicates that they are statistical significant with respect to the lead-3 baseline.

Table 4.1 Results on the Dailymail test samples

Method	75 bytes			275 bytes			
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L	F-1
Random 3	20.96	8.26	11.78	35.9	13.8	28.67	37.14
Lead 3	21.88	8.37	12.17	39.18	15.97	31.35	49.11
Topic Cluster	22.6*	9.49*	12.91*	38.47	15.7	31.22	55.65
Sentence Salience	22.27*	8.93*	12.53*	39.45*	16.2*	32.35*	59.56
Cheng and Lapata '16	22.7	8.5	12.5	42.2	**17.3**	34.8	-
Nallapati et al. '17	**26.2**	**10.7**	**14.4**	**42.2**	16.8	**35**	-

Table 4.2 Example output of the system generated summary - summaries are truncated to 275 bytes to indicate length of the summary considered by the ROUGE.

Gold Summary	Musk unveiled Powerwall device at press conference in California Daily use version will be able to store 7 kilowatt-hours of electricity It will let users store renewable energy, or pay lower, off-peak rates Also revealed a larger model which is a 'infinitely scalable
System Summary	Musk introduced the Powerwall device at a press conference in California last night and said the technology could 'change the world'. The device, which could be in homes by the end of summer, will be able to store electricity at night when it is cheaper. Scroll do

4.2 New York Times

The sentence salient model was trained on the newly introduced New York Times dataset. Our model performs close to the lead-3. The results for this model are presented in table

4.3. The lead-3 baseline is more competitive on this dataset because content in the human written reference abstracts are more strongly correlated with the first three sentences of the article than in the Dailymail corpus. F-1 score for the lead-3 baseline was also high when compared to the Dailymail corpus, further reflecting the high concentration of salient content in the article's opening paragraph. As a result, the model learns to give a high importance to the sentence position. Analysis of the predicted output labels showed that almost all of the included sentences were always from the first 4 sentences of the article. This indicates that the baseline for the New York Times is difficult to beat as the models would tend to give preference to the positional aspects of the sentence. A topic cluster based model trained on the NYTimes dataset performed significantly lower than the lead-3 sentences, suggesting that the coverage of the abstractive summaries is extremely low.

Table 4.3 Results on the New York Times test samples.

Method	75 bytes			275 bytes			
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L	F-1
Random 3	30.28	21.95	28.96	42.26	27.2	37.63	33.44
Lead 3	44.47	36.52	43.08	55.22	42.71	50.95	60.75
Sentence salience	44.18	36.21	42.79	54.96	42.38	50.76	60.77
Topic cluster	38.39	30.07	36.94	50.78	37.28	46.76	49.57

4.3 Document Understanding Conference 2002

A sentence salient based model trained on the Dailymail Corpus was tested with the DUC 2002 dataset. The results are presented in table 4.3. There are a total 567 articles in the DUC 2002 dataset. The results show that our model is significantly better than the lead-3 sentence summaries but does not outperform Cheng and Lapata, 2016 as the length of the summaries is considerably longer. We were not able to produce the same lead-3 scores as mentioned in Cheng and Lapata, 2016 even after trying a variety of different ROUGE settings. The fact that we could not replicate their scores exactly and as our scores for lead-3 were slightly lower, leaves open the possibility that overall their scores can be slightly inflated over ours. Our ROUGE scores are calculated at the length of 75 words with stemming and without the removal of stop words. The state of the art on this dataset is given by URANK (Wan and Xiao) and TGRAPH (Parveen et al.). * represents that the model is statistically significant with respect to the leading 3 sentences.

Table 4.4 Results on DUC 2002

Method	75 words		
	Rouge-1	Rouge-2	Rouge-L
Lead 3	42.72	21.32	38.57
Cheng et al '16 Lead 3	43.6	21.0	40.2
Sentence salience	45.87*	22.45*	41.48*
Cheng et al '16	47.4	23.0	**43.5**
Nallapati et al. '17	46.8	22.6	43.1
URANK	**48.5**	21.5	-
TGRAPH	48.1	**24.3**	-

Chapter 5

Future Work and Conclusion

5.1 Future Work

For future work, we plan to replace the named entities related to persons, places, dates, etc. with a unique token for each category. The presence of named entities in the sentence is an important feature for sentence selection. But named entities don't occur very frequently. In the current setting where all the words which occur less than 50 times throughout the corpus are replaced with an unknown token, we lose the less frequent named entities with other noisy words. We also plan to implement a new model which makes use of the headline of the article while performing extractive summarization. Important sentences in the article have a high correlation with the headline of the article. Adding a term about the similarity between the headline and sentence of the article in the energy equation will help in performing better extractive summaries.

In our models, we have truncated the articles to the first 10 sentences due to the resource constraints. On one hand, this may have given our models an advantage because it reduces the search space for the task as the top 10 sentences are more likely to be in the summary. But this may have also hurt our models because there could be sentences with high importance appearing at the end of the article that are ignored by our models. In the future, we would like to consider more sentences for each article and compare their performance.

5.2 Conclusion

In this work we have presented two models to perform extractive summarization of news articles. These models are based on Structured Prediction Energy Networks. Each of the models tries to capture different concepts important to the summarization task. One of them

is based on sentence salience and another one is based on the topic clustering. The sentence salience model performs better than the topic clustering model. One of our hypothesis for the lower performance of the topic clustering model is due to lower variation in the topics discussed in a news article. Experimental results show that these models are good at short length summaries but lags behind on the longer length summaries. As our models perform prediction for all the labels simultaneously, it is difficult for it to reduce the redundancies in the summaries. Results also show that the sentence salience based model performs better than the strong lead-3 sentence baseline. We also introduced recent New York Times articles to the New York Times Annotated Corpus. It was found that New York Times summaries have a strong correlation with the leading sentences of the article. As a result, when the sentence salient based model was trained on it, it gave a high preference to the starting position and performed similar to the lead-3 sentences. Analysis of the predicted summaries showed that they are mostly comprised of the first few sentences of the article. Overall, we have contributed in presenting two novel architectures for the extractive summarization, introducing a new large scale corpus and using our system in another project.

References

[1] The new york times annotated corpus linguistic data consortium, philadelphia, vol. 6, no. 12. (2008) by e. sandhaus.

[2] Pretrained word2vec embeddings. URL https://code.google.com/archive/p/word2vec/.

[3] David Belanger and Andrew McCallum. Structured prediction energy networks. In *Proceedings of the International Conference on Machine Learning*, 2016.

[4] Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words. *arXiv preprint arXiv:1603.07252*, 2016.

[5] Günes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, 2004.

[6] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701, 2015.

[7] Yoon Kim. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882, 2014. URL http://arxiv.org/abs/1408.5882.

[8] Eliyahu Kiperwasser and Yoav Goldberg. Simple and accurate dependency parsing using bidirectional LSTM feature representations. *CoRR*, abs/1603.04351, 2016. URL http://arxiv.org/abs/1603.04351.

[9] Julian Kupiec, Jan Pedersen, and Francine Chen. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM, 1995.

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

[11] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[12] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *CoRR*, abs/1611.04230, 2016. URL http://arxiv.org/abs/1611.04230.

[13] Ani Nenkova. Automatic text summarization of newswire: Lessons learned from the document understanding conference. In *AAAI*, volume 5, pages 1436–1441, 2005.

[14] Daraksha Parveen, Hans martin Ramsl, and Michael Strube. Topical coherence for graph-based extractive summarization.

[15] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *CoRR*, abs/1509.00685, 2015. URL http://arxiv.org/abs/1509.00685.

[16] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215, 2014. URL http://arxiv.org/abs/1409.3215.

[17] Xiaojun Wan and Jianguo Xiao. Exploiting neighborhood knowledge for single document summarization and keyphrase extraction. *ACM Trans. Inf. Syst.*, 28(2): 8:1–8:34, June 2010. ISSN 1046-8188. doi: 10.1145/1740592.1740596. URL http://doi.acm.org/10.1145/1740592.1740596.

[18] Kristian Woodsend and Mirella Lapata. Automatic generation of story highlights. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 565–574. Association for Computational Linguistics, 2010.