Neural Summarization of documents by extracting sentences

Guided by:
Abhishek Singh

Submitted by:
Group8
Suryansh Agnihotri
Anandita Dhasmana
Piyush Shrivastava
Shalin Agarwal

Introduction

- **Problem statement:single document summarization by extracting sentences using deep learning.**

- Model used:- A neural network-based hierarchical document reader or encoder and an attention based content extractor is used.

- Traditional approaches to extractive summarization rely heavily on human-engineered features

- We have used a data-driven approach based on neural networks and continuous sentence features.

- A general framework for single-document summarization composed of a hierarchical document encoder and an attention-based extractor is introduced.

- This architecture allows us to develop different classes of summarization models which can extract sentences or words.
- The role of the reader is to derive the meaningful representation(encoding) of a document based on its sentences and their constituent words.
- Sentence extractor uses attention mechanism for sequence labeling.

Training Data

- DailyMail news dataset is used.

- Dataset division:
  90% of data set for training
  5% for validation
  5%for testing

```
as the model for @entity4 's " @entity3 , " @entity1 became the symbol of @entity7 women working on the home front during @entity9 the 92 - year - old died this week
at her home in @entity12 , @entity13                    1
as a 19 - year - old telephone operator , @entity1 posed for the famous painting that would become the cover of the @entity17 on may 29 , 1943                    1
although she was petite , @entity1 was transformed into the iconic -- and burly -- embodiment of the character by @entity4                    1
" other than the red hair and my face , @entity4 embellished @entity26 's body , " @entity1 said in a 2012 interview with the @entity22                    1
" i was much smaller than that and did not know how he was going to make me look like that until i saw the finished painting                    0
" people we 've lost in 2015 @entity1 pocketed $ 10 for the two mornings of modeling work she did in @entity34 , @entity35                    2
@entity4 lived in neighboring @entity34 at the time                    0
" @entity3 " is often confused with another popular image from the same era                    2
the poster shows a woman flexing her arm under the slogan " @entity42                    0
" it was part of a nationwide campaign to sell war bonds , but is not the same character                    0
still , many folks on social media paid tribute to @entity1 using the image                    2
both show the key role women played in the war effort .                    0

" @entity3 " appeared on the cover of the @entity17 on may 29 , 1943
@entity1 was a 19 - year - old telephone operator at the time

@entity3:Rosie the Riveter
@entity17:Saturday Evening Post
@entity1:Mary Doyle Keefe
@entity0:CNN
```

MATHEMATICAL FORMULATION

Given a document D consisting of a sequence of sentences $\{s\ 1\ ,\ \cdot\ \cdot\ \cdot\ ,\ s\ m\ \}$ Sentence extraction aims to create a summary from D by selecting a subset of j sentences.
predicting a label $y_L \in \{0, 1\}$

<u>Objective function</u>:-

Given the input document D and model parameters θ:

$$\log p(y_L|D; \theta) = \sum \log p(y^i_L|D; \theta)$$

NEURAL SUMMARIZATION MODEL

- The key components of our summarization model include a neural network-based hierarchical document reader and an attention-based hierarchical content extractor.

- Such a representation yields minimum information loss and is flexible allowing us to apply neural attention for selecting salient sentences.

- Key components used:
  -neural network-based hierarchical document reader
  -attention -based hierarchical content extractor

DOCUMENT READER

- The role of the reader is to derive the meaning representation of the document from its constituent sentences, each of which is treated as a sequence of words.

Convolutional Sentence Encoder

sum these sentence vectors

$$\mathbf{s}_{i,\mathbf{K}} = \max_j \mathbf{f}^i_j \qquad (4)$$

$$\mathbf{f}^i_j = \tanh(\mathbf{W}_{j:j+c-1} \otimes \mathbf{K} + b) \qquad (3)$$

kernel $\mathbf{K} \in R^{c \times d}$ of width c

$\mathbf{W} \in R^{n \times d}$

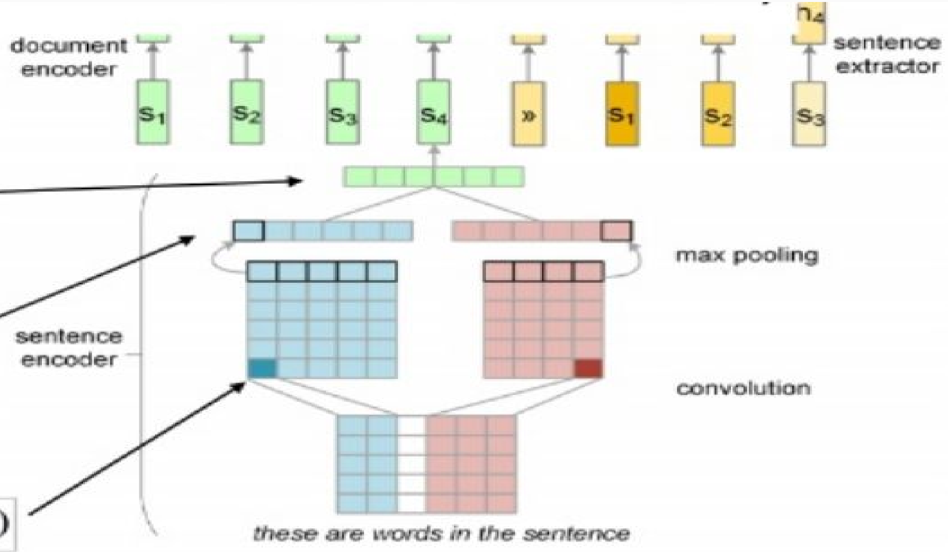
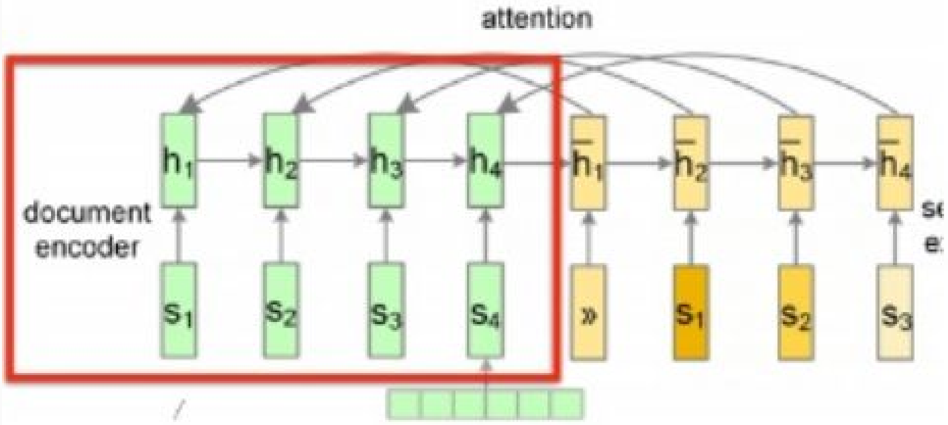


Figure 2: A recurrent convolutional document reader with a neural sentence extractor.

Long Short-Term Memory (LSTM) activation unit is used in the RNN for ameliorating the vanishing gradient problem when training long sequences.

Recurrent Document Encoder:-



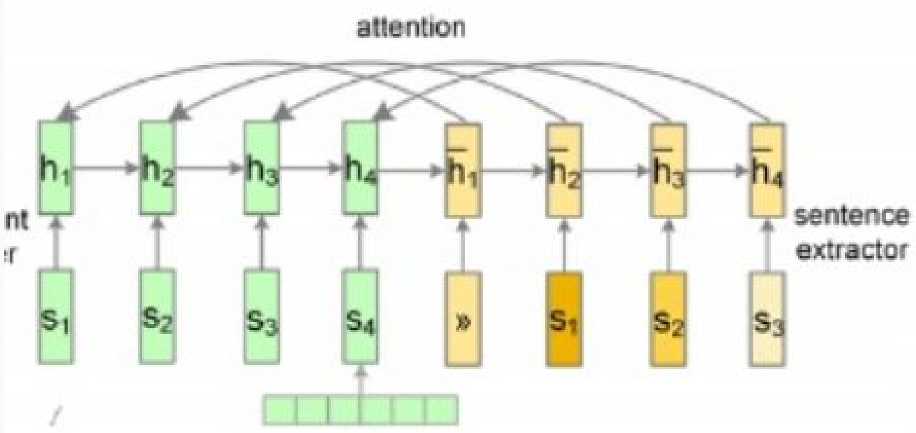
$$\begin{bmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \hat{\mathbf{c}}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} \mathbf{W} \cdot \begin{bmatrix} \mathbf{h}_{t-1} \\ \mathbf{s}_t \end{bmatrix}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$$

SENTENCE EXTRACTOR

- sentence extractor applies attention to directly salient sentences after reading them
- Set Pt-1 to the true label of the previous sentence as training goes on they gradually shift its value to the predicted label.



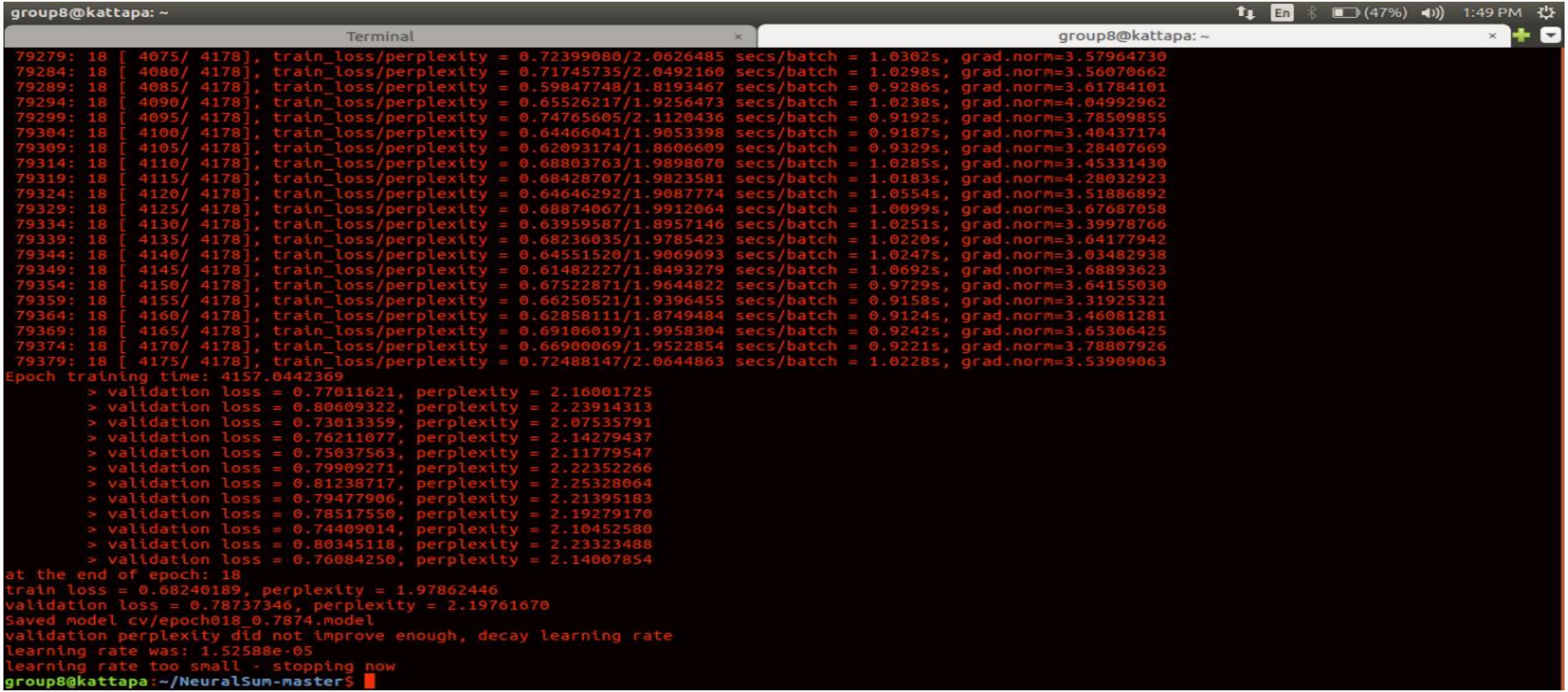
$$\bar{\mathbf{h}}_t = \mathrm{LSTM}(p_{t-1}\mathbf{s}_{t-1}, \bar{\mathbf{h}}_{t-1})$$

$$p(y_L(t) = 1|D) = \sigma(\mathrm{MLP}(\bar{\mathbf{h}}_t : \mathbf{h}_t))$$

10

Result

Model was trained in 18 epoch of the training dataset:



Score generation for Test data:

Sentence scores are generated and stored

```
9.949956494383513927e-01 7.101479917764663696e-01 7.060083001852035522e-01 1.238418444991111755e-01 4.624989330768585205e-01 1.993201747536659241e-01
3.840281814336776733e-01 7.199154645204544067e-01 2.584178000688552856e-01 3.626094162464141846e-01 5.996743440628051758e-01 1.217002160847187042e-01
5.549399703741073608e-01 3.994210436940193176e-01 1.621807896867721865e-07
9.803755236789584160e-01 9.883490633219480515e-01 9.616373460739850998e-01 4.649035930633544922e-01 8.931522220373153687e-01 9.598650056868791580e-01
8.513485267758369446e-02 4.951936900615692139e-01 6.704502403736114502e-01 4.238985180854797363e-01 2.506899759173393250e-01 4.282903745770454407e-01
1.948903873562812805e-01 1.339862719178199768e-01 5.209758654236793518e-01
9.470038060098886490e-01 3.329327851533889771e-01 8.667793422937393188e-01 3.271220773458480835e-01 8.187378868460655212e-01 9.322041794657707214e-01
3.561458736658096313e-01 9.533735625445842743e-01 9.108157195150852203e-01 8.284008875489234924e-01 1.474819257855415344e-01 2.700386419892311096e-01
7.888400629162788391e-01 2.834613621234893799e-01 8.629281520843505859e-01
9.740489851683378220e-01 9.155891425907611847e-01 3.406101912260055542e-01 2.958291769027709961e-01 5.060718208551406860e-01 4.430162720382213593e-02
5.825996696949005127e-01 3.519710898399353027e-01 5.402918159961700439e-01 4.129633903503417969e-01 4.437561333179473877e-01 4.655369818210601807e-01
8.468578308820724487e-01 8.318230584263801575e-01 8.672466501593589783e-02
9.913309137336909771e-01 9.209049306809902191e-01 9.734167642891407013e-01 8.478100746870040894e-01 2.522733360528945923e-01 3.865976333618164062e-01
9.415631815791130066e-01 5.875742435455322266e-01 8.117417097091674805e-01 1.138487346470355988e-01 7.902673035860061646e-01 1.874222531914710999e-01
3.138754889369010925e-01 5.088656246662139893e-01 4.671209752559661865e-01
```

Sentence scores are stored during evaluation.

	sentence1	sentence2				
doc1	8.42e-01	8.13e-01	8.37e-01	6.27e-01	5.07e-01	6.57e-01
doc2	9.12e-01	7.97e-01	8.03e-01	7.61e-01	8.85e-01	7.47e-01
	9.07e-01	8.68e-01	9.13e-01	9.46e-01	7.22e-01	9.21e-01

CONCLUSION

- We developed a model based on sentence extraction.
- This architecture can further be extended for generating summaries in english using the word extractor.

THANK YOU