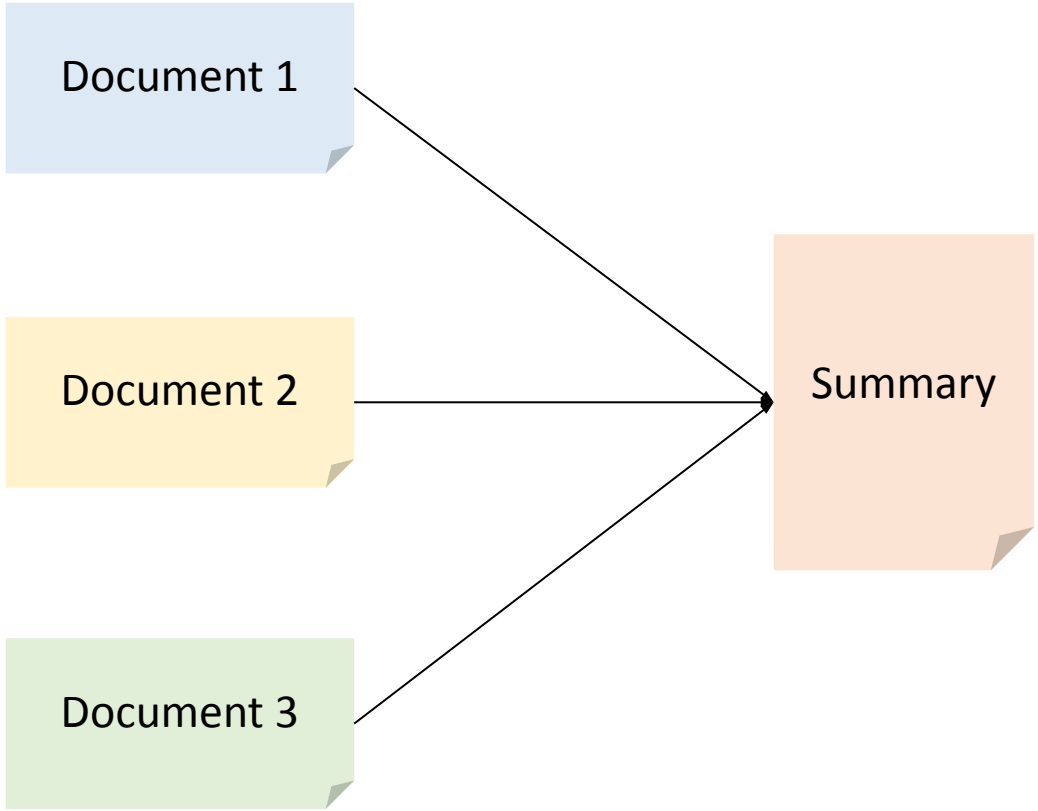
Advanced NLP

Multi-Document Summarization

Yixin Liu (yixin.liu@yale.edu)
September 30, 2021

Introduction

- Combines information from multiple documents into a coherent summary



Source 1
Meng Wanzhou, Huawei's chief financial officer and deputy chair, was arrested in Vancouver on 1 December. Details of the arrest have not been released...
Source 2
A Chinese foreign ministry spokesman said on Thursday that Beijing had separately called on the US and Canada to "clarify the reasons for the detention "immediately and "immediately release the detained person ". The spokesman...
Source 3
Canadian officials have arrested Meng Wanzhou, the chief financial officer and deputy chair of the board for the Chinese tech giant Huawei,...Meng was arrested in Vancouver on Saturday and is being sought for extradition by the United States. A bail hearing has been set for Friday...
Summary
...Canadian authorities say she was being sought for extradition to the US, where the company is being investigated for possible violation of sanctions against Iran. Canada's justice department said Meng was arrested in Vancouver on Dec. 1... China's embassy in Ottawa released a statement.. "The Chinese side has lodged stern representations with the US and Canadian side, and urged them to immediately correct the wrongdoing "and restore Meng's freedom, the statement said...

(Fabbri, et al. 2019)

Introduction

- Importance
  - Sea of information on the internet – tl;dr
  - Application Scenarios
    - News Articles Summarization
    - Survey Generation
    - …
- Challenges (compared with single document summarization)
  - Redundancy
  - Conflicting Information
  - Long Text

Earlier Work

Journal of Artificial Intelligence Research 22 (2004) 457-479 Submitted 07/04; published 12/04

LexRank: Graph-based Lexical Centrality as Salience in Text Summarization

Güneş Erkan GERKAN@UMICH.EDU
Department of EECS
University of Michigan, Ann Arbor, MI 48109 USA

Dragomir R. Radev RADEV@UMICH.EDU
School of Information & Department of EECS
University of Michigan, Ann Arbor, MI 48109 USA

- LexRank: Graph-based Lexical Centrality as Salience in Text Summarization (Erkan et al., 2004)
  - Multi-document
  - Extractive Summarization
    - *"Extractive summarization produces summaries by choosing a subset of the sentences in the original document(s)"*
    - Abstractive Summarization – *"the information in the text is rephrased"*
  - Graph-based Algorithm – for computing the sentence importance

Earlier Work - LexRank

"Identifying the most central sentences in a (multidocument) cluster that give the necessary and sufficient amount of information related to the main theme of the cluster"

$$\text{idf-modified-cosine}(x, y) = \frac{\sum_{w \in x,y} \text{tf}_{w,x} \text{tf}_{w,y} (\text{idf}_w)^2}{\sqrt{\sum_{x_i \in x} (\text{tf}_{x_i,x} \text{idf}_{x_i})^2} \times \sqrt{\sum_{y_i \in y} (\text{tf}_{y_i,y} \text{idf}_{y_i})^2}}$$

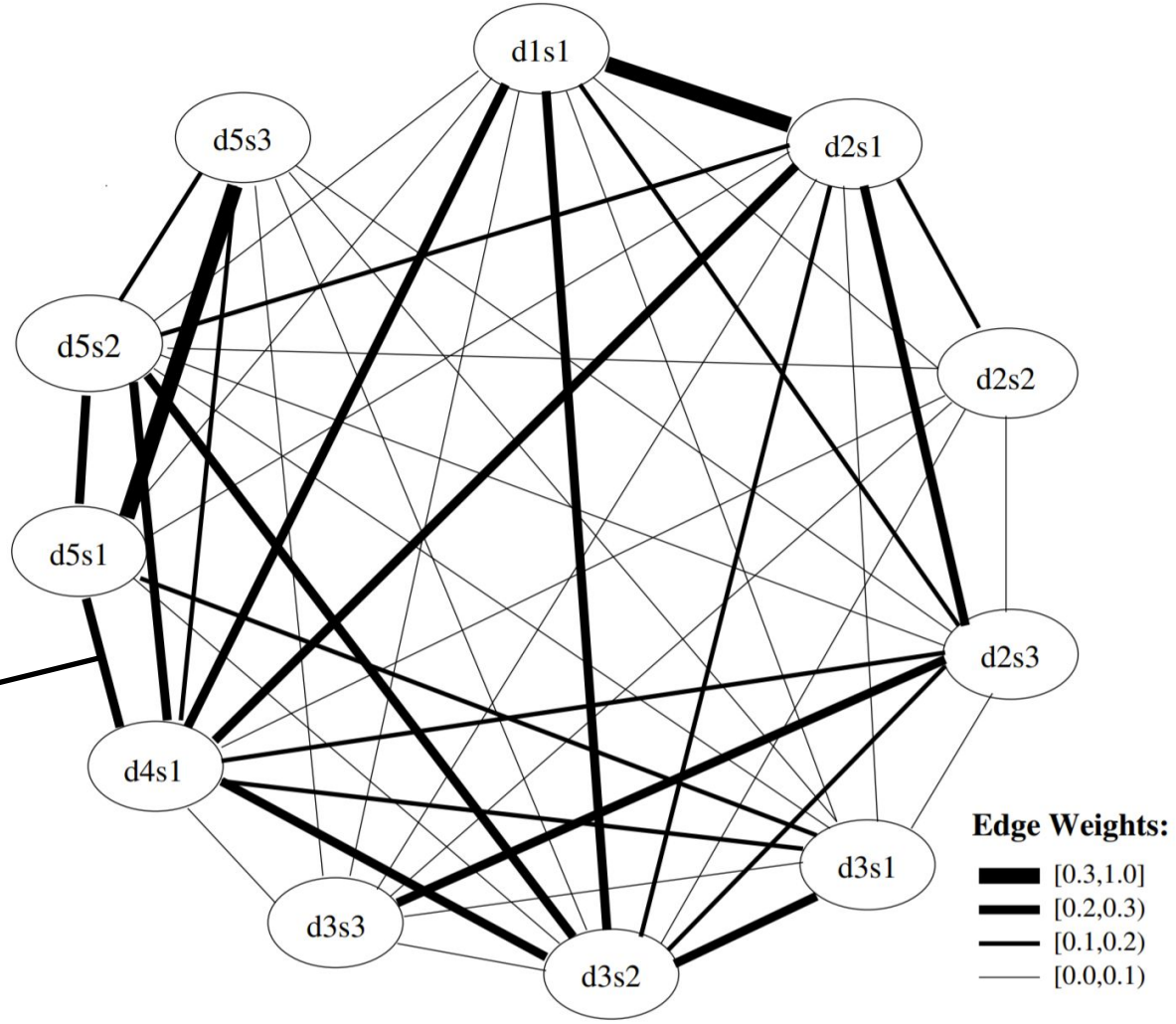


Figure 2: Weighted cosine similarity graph for the cluster in Figure 1.

Earlier Work - LexRank

```
1  MInputAn array S of n sentences, cosine threshold t  output: An array L of LexRank scores
2  Array CosineMatrix[n][n];
3  Array Degree[n];
4  Array L[n];
5  for i ← 1 to n do
6      for j ← 1 to n do
7          CosineMatrix[i][j] = idf-modified-cosine(S[i],S[j]);
8          if CosineMatrix[i][j] > t then
9              CosineMatrix[i][j] = 1;
10             Degree[i] + +;
11         end
12         else
13             CosineMatrix[i][j] = 0;
14         end
15     end
16 end
17 for i ← 1 to n do
18     for j ← 1 to n do
19         CosineMatrix[i][j] = CosineMatrix[i][j]/Degree[i];
20     end
21 end
22 L = PowerMethod(CosineMatrix,n,ϵ);
23 return L;
```

Algorithm 3: Computing LexRank scores.

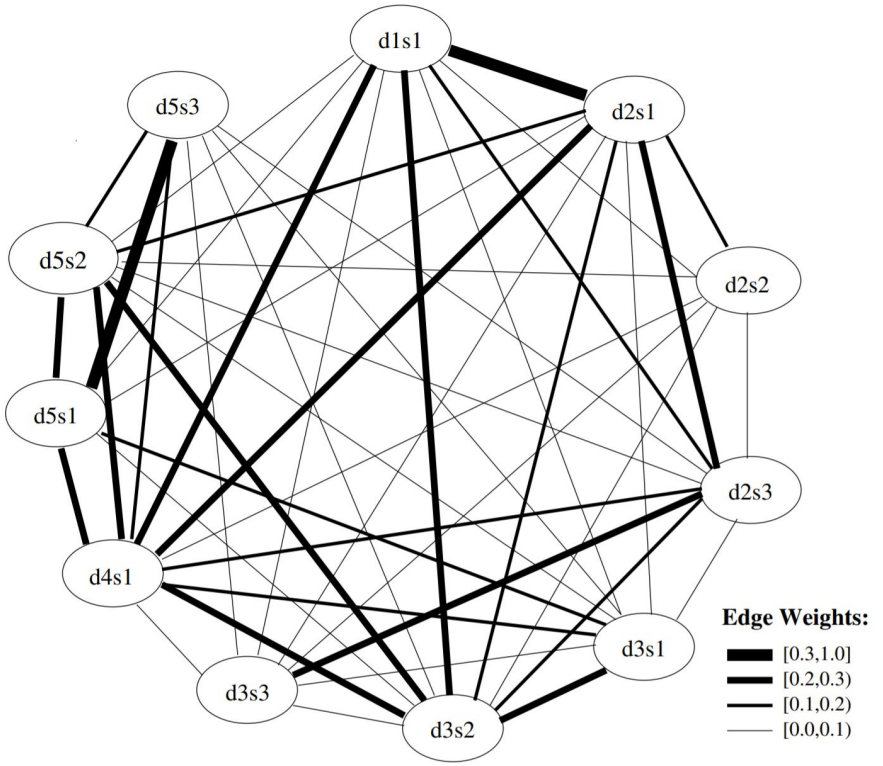


Figure 2: Weighted cosine similarity graph for the cluster in Figure 1.

Earlier Work

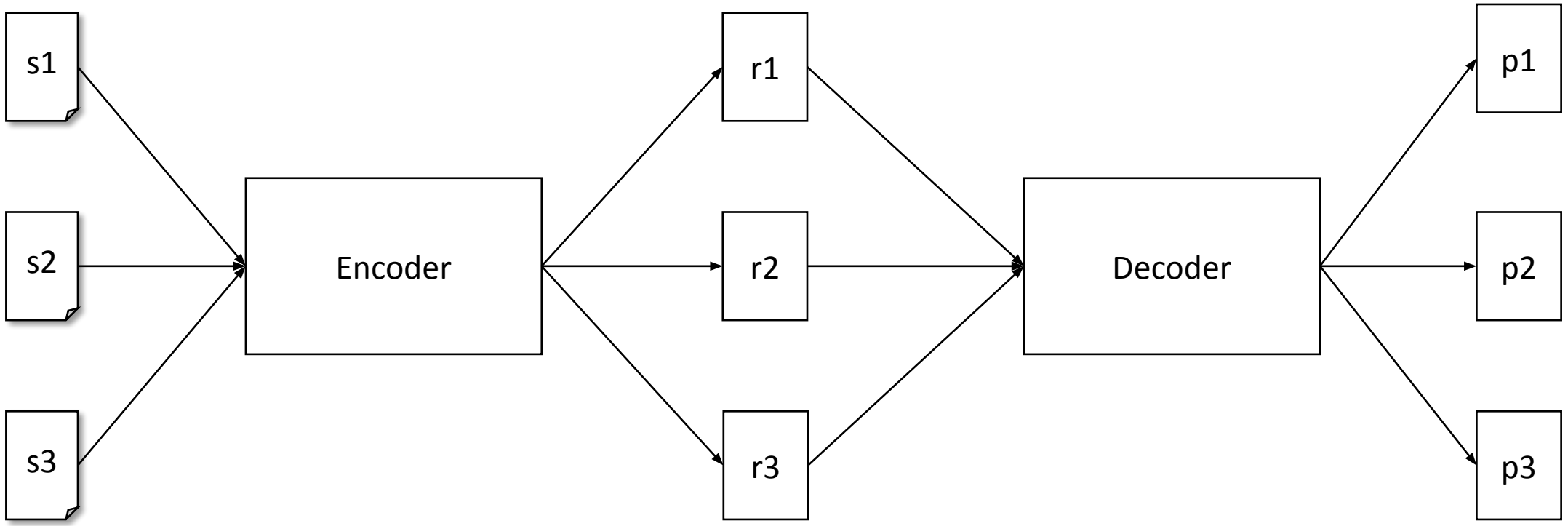
Graph-based Neural Multi-Document Summarization

Michihiro Yasunaga[1] **Rui Zhang**[1] **Kshitijh Meelu**[1]
Ayush Pareek[2] **Krishnan Srinivasan**[1] **Dragomir Radev**[1]
[1]Department of Computer Science, Yale University
[2]The LNM Institute of Information Technology
{michihiro.yasunaga,r.zhang,kshitijh.meelu}@yale.edu
{ayush.original}@gmail.com
{krishnan.srinivasan,dragomir.radev}@yale.edu

- Graph-based Neural Multi-document Summarization (Yasunaga et al., 2017)
  - Multi-document
  - Neural Extractive Summarization
  - Graph Convolutional Networks

Earlier Work – Graph-based Neural MDS

- General Pipeline of Neural Extractive Summarization



Earlier Work – Graph-based Neural MDS

- Graph-based Encoding

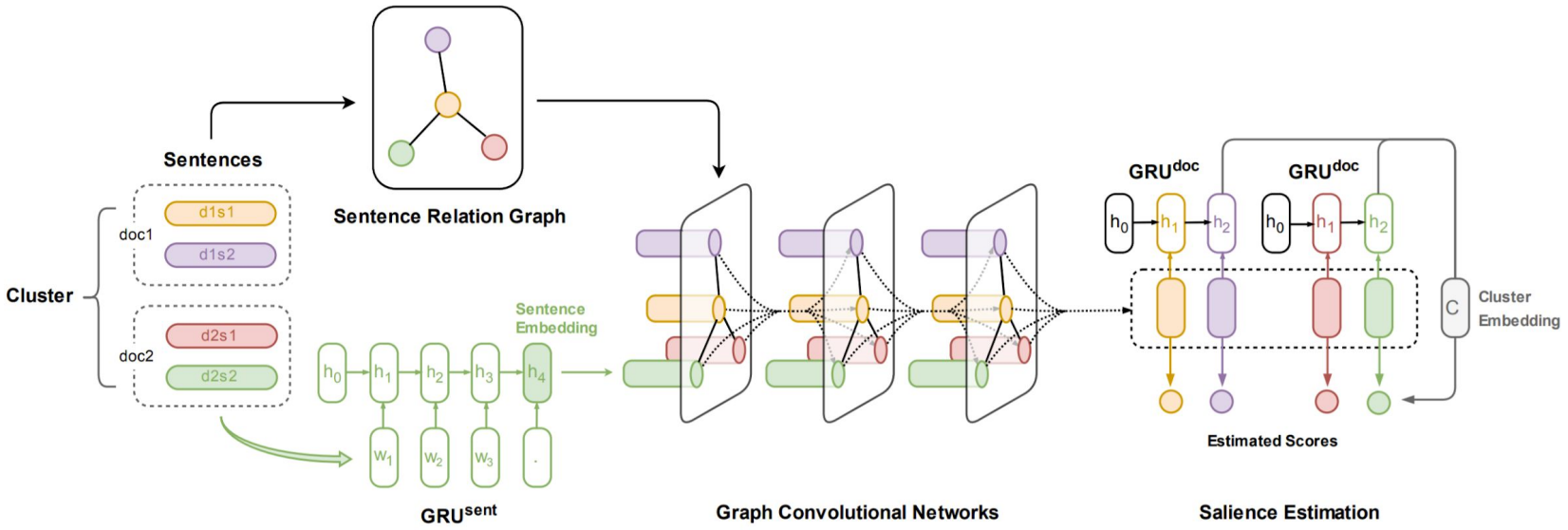


Figure 1: Illustration of our architecture for sentence salience estimation. In this example, there are two documents in the cluster and each document has two sentences. Sentences are processed by the GRU^{sent} to get input sentence embeddings. The GCN takes the input sentence embeddings and the sentence relation graph, and outputs high-level hidden features for individual sentences. GRU^{doc} produces the cluster embedding from the output sentence embeddings. The salience is estimated from the output sentence embeddings and the cluster embedding. w_i: the word embedding for i-th word. h_i: the hidden state of GRU at i-th step.

Earlier Work – Graph-based Neural MDS

- Graph-based Encoding

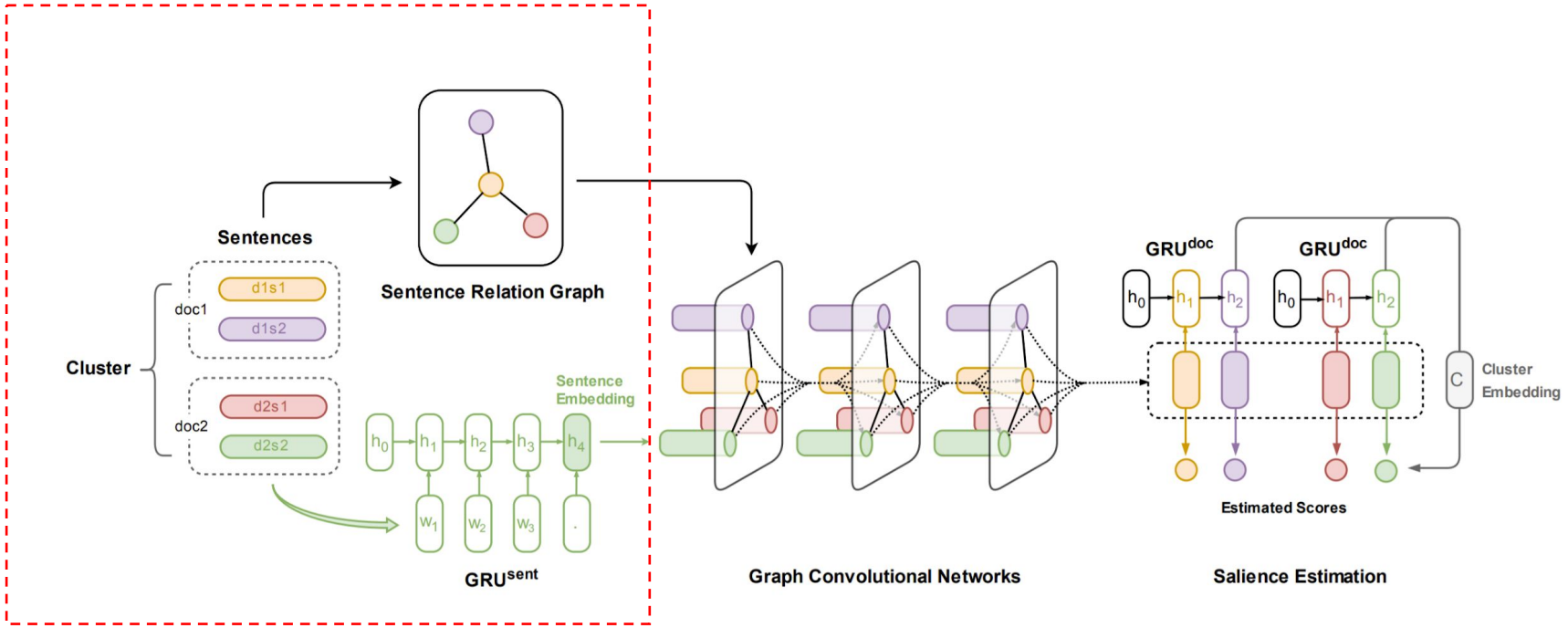


Figure 1: Illustration of our architecture for sentence salience estimation. In this example, there are two documents in the cluster and each document has two sentences. Sentences are processed by the GRU^{sent} to get input sentence embeddings. The GCN takes the input sentence embeddings and the sentence relation graph, and outputs high-level hidden features for individual sentences. GRU^{doc} produces the cluster embedding from the output sentence embeddings. The salience is estimated from the output sentence embeddings and the cluster embedding. w_i: the word embedding for i-th word. h_i: the hidden state of GRU at i-th step.

Earlier Work – Graph-based Neural MDS

- Graph-based Encoding

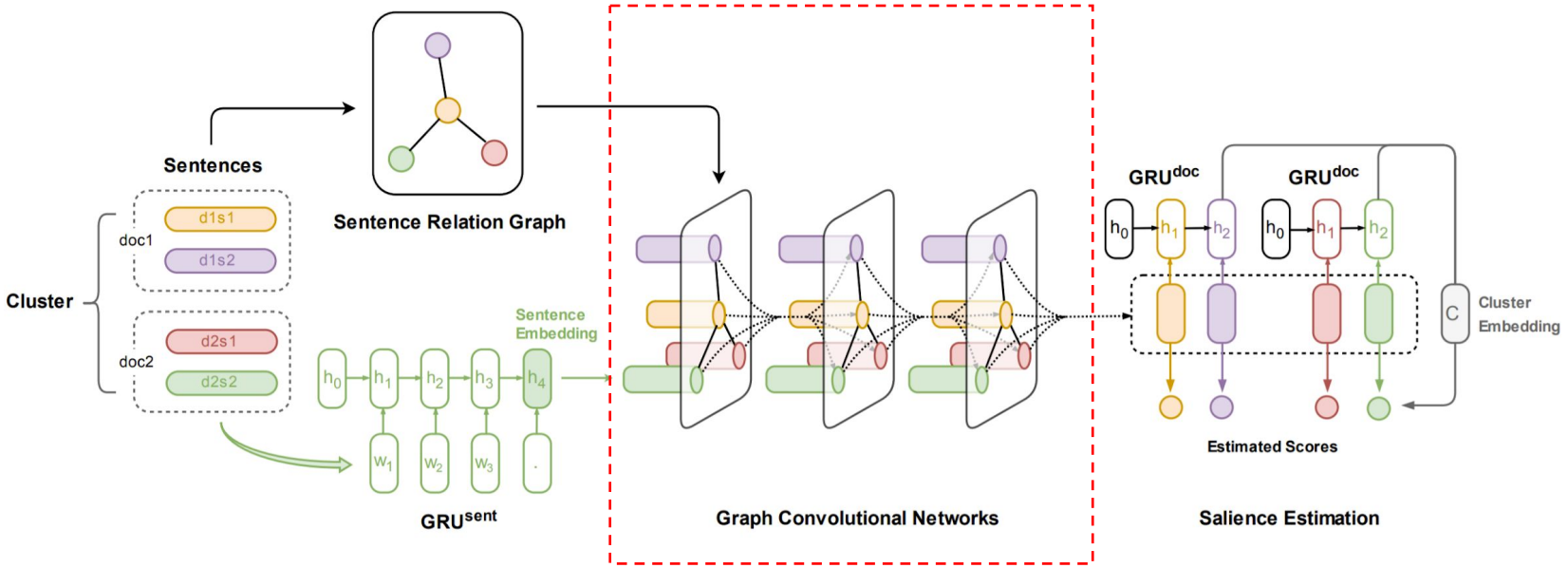


Figure 1: Illustration of our architecture for sentence salience estimation. In this example, there are two documents in the cluster and each document has two sentences. Sentences are processed by the GRU^{sent} to get input sentence embeddings. The GCN takes the input sentence embeddings and the sentence relation graph, and outputs high-level hidden features for individual sentences. GRU^{doc} produces the cluster embedding from the output sentence embeddings. The salience is estimated from the output sentence embeddings and the cluster embedding. w_i: the word embedding for i-th word. h_i: the hidden state of GRU at i-th step.

Earlier Work – Graph-based Neural MDS

- Graph-based Encoding

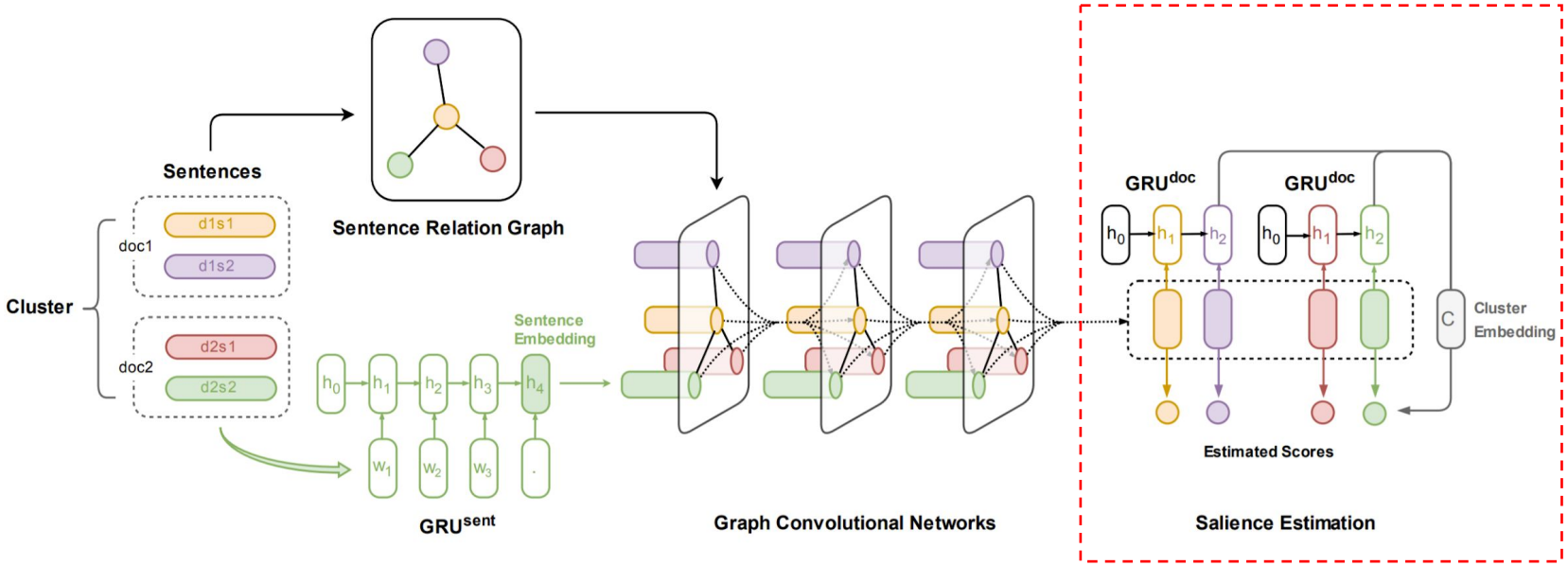
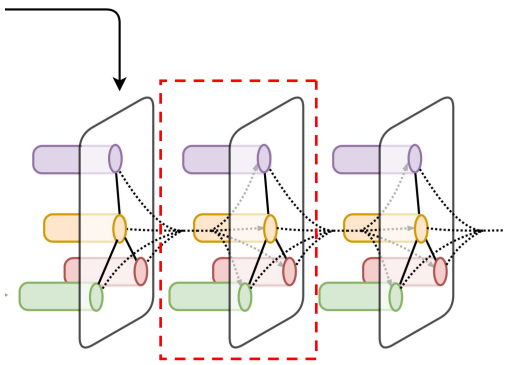


Figure 1: Illustration of our architecture for sentence salience estimation. In this example, there are two documents in the cluster and each document has two sentences. Sentences are processed by the GRU^{sent} to get input sentence embeddings. The GCN takes the input sentence embeddings and the sentence relation graph, and outputs high-level hidden features for individual sentences. GRU^{doc} produces the cluster embedding from the output sentence embeddings. The salience is estimated from the output sentence embeddings and the cluster embedding. w_i: the word embedding for i-th word. h_i: the hidden state of GRU at i-th step.

Earlier Work – Graph-based Neural MDS

- Graph Convolutional Networks – Computing the node representations in a relation graph

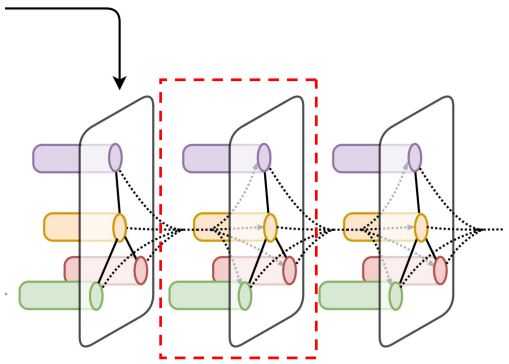
  - $A \in \mathbb{R}^{N \times N}$, the adjacency matrix of graph $\mathcal{G}$, where N is the number of nodes in $\mathcal{G}$.
  - $X \in \mathbb{R}^{N \times D}$, the input node feature matrix, where D is the dimension of input node feature vectors.



Graph Convolutional Networks

Earlier Work – Graph-based Neural MDS

- Graph Convolutional Networks – Computing the node representations in a relation graph
  - $A \in \mathbb{R}^{N \times N}$, the adjacency matrix of graph $\mathcal{G}$, where N is the number of nodes in $\mathcal{G}$.
  - $X \in \mathbb{R}^{N \times D}$, the input node feature matrix, where D is the dimension of input node feature vectors.



Graph Convolutional Networks

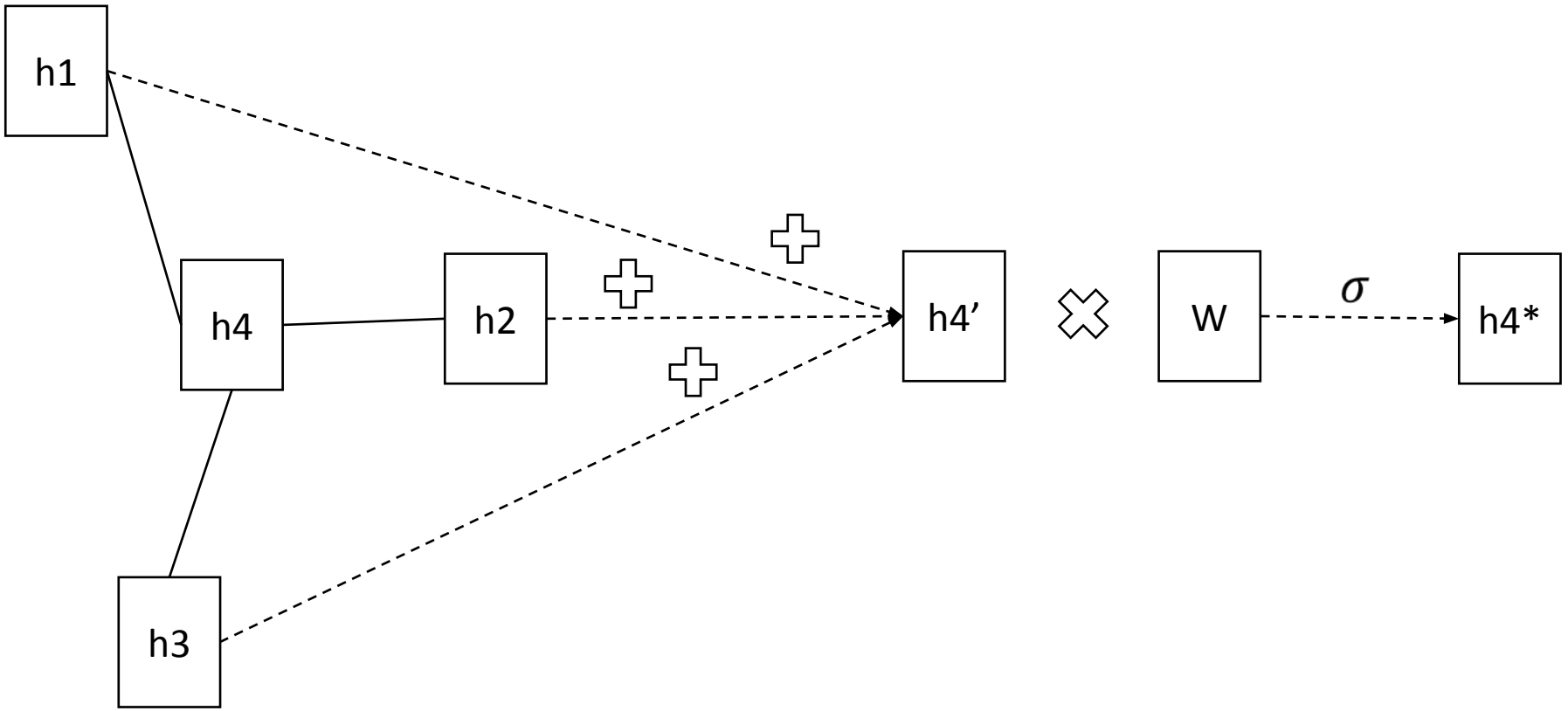
- Later-wise Computation

$$H^{(l+1)} = \sigma\left(AH^{(l)}W^{(l)}\right)$$

Earlier Work – Graph-based Neural MDS

- Graph Convolutional Network

$$H^{(l+1)} = \sigma\left(AH^{(l)}W^{(l)}\right)$$



Earlier Work – Graph-based Neural MDS

- Relation Graph
  - tf-idf cosine similarity
  - Approximate Discourse Graph (ADG) – counting discourse relation indicators
    - Discourse markers
    - Coreference Chain
    - Deverbal noun references
  - Personalized ADG

$$w_{PDG}(u,v) = \frac{w_{ADG}(u,v)s(v)}{\sum_{u' \in V} w_{ADG}(u',v)s(u')}$$

Personalization Features
• Position in Document
• From 1^{st} 3 Sentences?
• No. of Proper Nouns
• > 20 Tokens in Sentence?
• Sentence Length
• No. of Co-referent Verb Mentions
• No. of Co-referent Common Noun Mentions
• No. of Co-referent Proper Noun Mentions

Datasets for MDS

1. DUC-2003/2004
   a. small datasets (< 100 clusters)
   b. relatively short summaries (< 100 words)

2. TAC-2008/2009/2010/2011

3. Multi-News (ACL 2019)
   a. the first large-scale multi-document news summarization dataset
   b. > 50k articles-summary pairs

4. Wikipedia Current Events Portal (ACL 2020)
   a. human written summaries of new events
   b. > 10k clusters, > 200 articles per cluster on average

5. Query-focused Multi-Document Summarization (AAAI 2021)
   a. QMDSCNN: converting CNN/DM dataset to a query-focused dataset, > 287k training samples
   b. QMDSIR: constructed from the search engine log of real users, > 80k training samples

Paper 1

Efficiently Summarizing Text and Graph Encodings of Multi-Document Clusters

Ramakanth Pasunuru[1] **Mengwen Liu**[2] **Mohit Bansal**[1]
Sujith Ravi[2] **Markus Dreyer**[2]
[1]UNC Chapel Hill
[2]Amazon Alexa
{ram,mbansal}@cs.unc.edu, {mengwliu,sujithai,mddreyer}@amazon.com

- Efficiently Summarizing Text and Graph Encodings of Multi-Document Clusters (**NAACL 2021**)
  - Neural Abstractive Summarization
  - Large Pre-trained Language Model (BART)
  - Sparse Attention Model (Longformer)
  - Linearized Graph Representation

Paper 1 - Motivation

- Limitations of the previous work
  - Not Scalable for long documents
    - Encoding length limit
    - Quadratic memory growth

Table 1: Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types. n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in restricted self-attention.

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

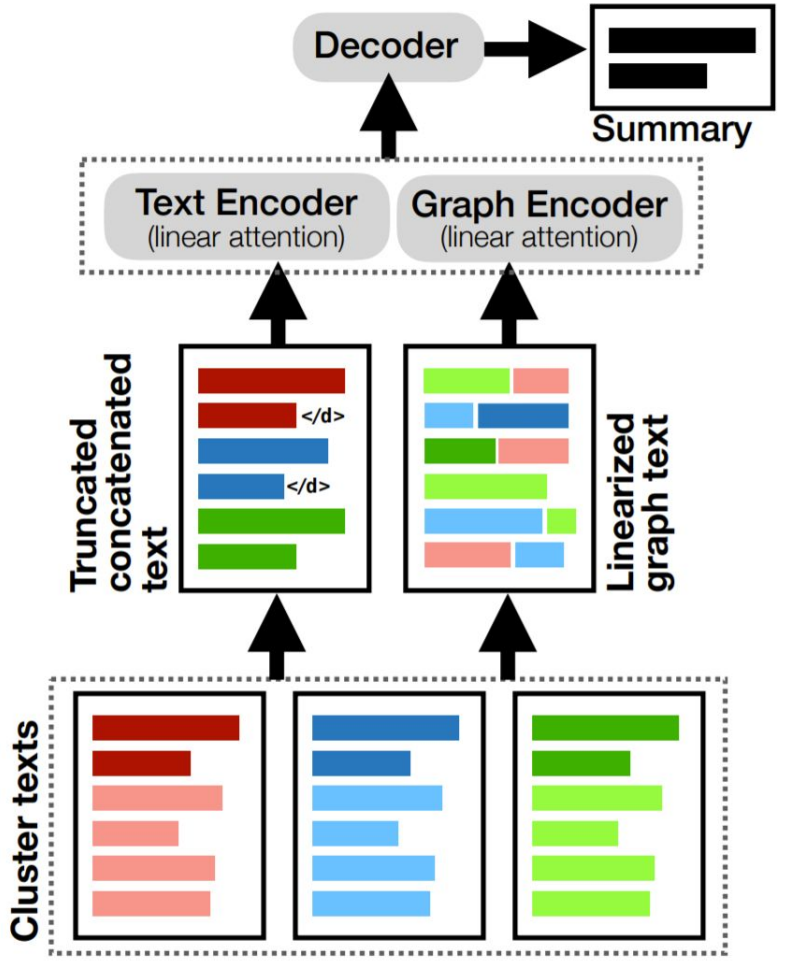
(Attention is all you need, Vaswani et al., 2017)

    - Example – maximum token length of BART: *1024* v.s. average input length of multinews: 2103

  - Do not jointly explore alternate auxiliary information

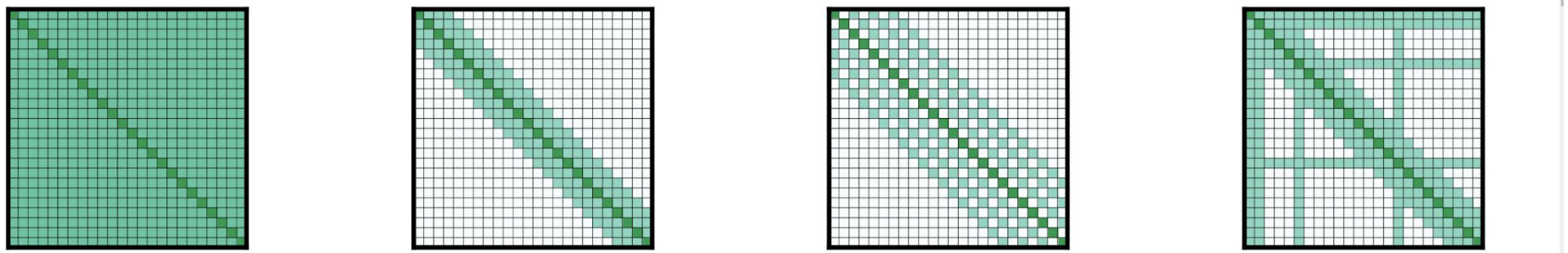
Paper 1 - Method

- Text Encoder
  - Input – Truncated Concatenated Text
- Graph Encoder
  - Input – Linearized Graph Text
- Decoder
  - Input – Combined Text and Graph Representations



Paper 1 - Method

- Longformer Attention
  - Longformer: The Long-Document Transformer (Beltagy et al., 2020)
  - Using global and sliding window attention

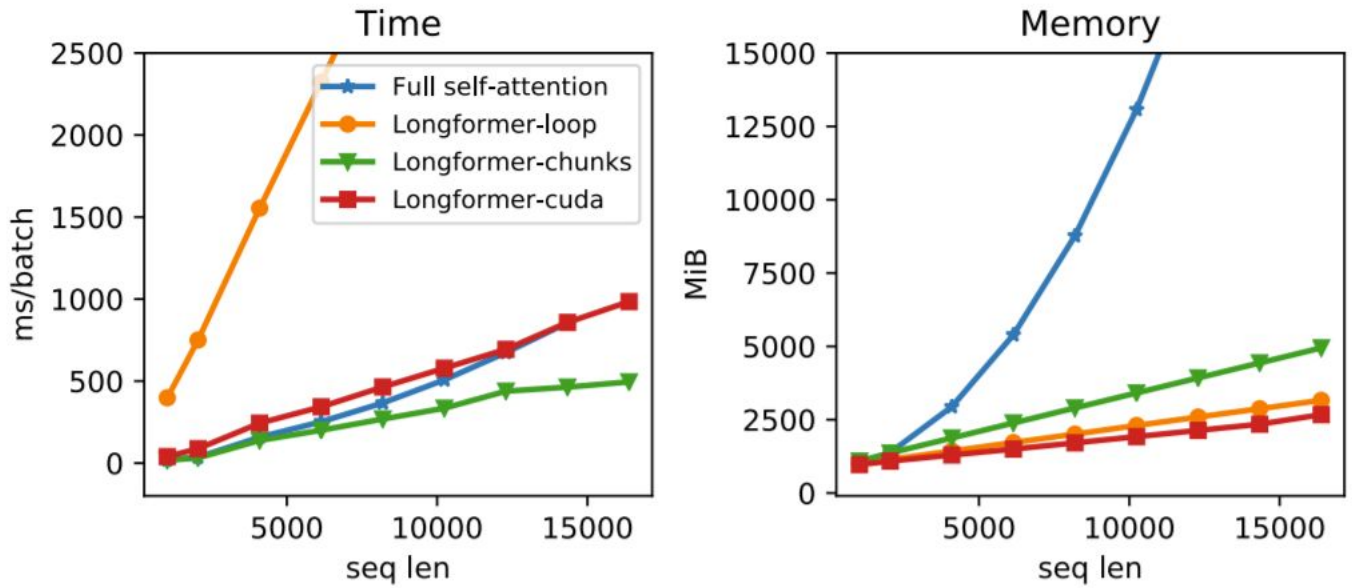


(a) Full n^2 attention (b) Sliding window attention (c) Dilated sliding window (d) Global+sliding window

Figure 2: Comparing the full self-attention pattern and the configuration of attention patterns in our Longformer.

Paper 1 - Method

- Longformer Attention
  - Longformer: The Long-Document Transformer (Beltagy et al., 2020)
  - Using global and sliding window attention
  - Scales linearly with sequence length



Paper 1 - Method

- Graph Construction
  - Coreference Resolution
  - Open Information Extraction Triplets (<sub> <pred> <obj>)
  - Linearized Graph Text
    - Starting with the node with highest centrality
    - Breadth-first Search

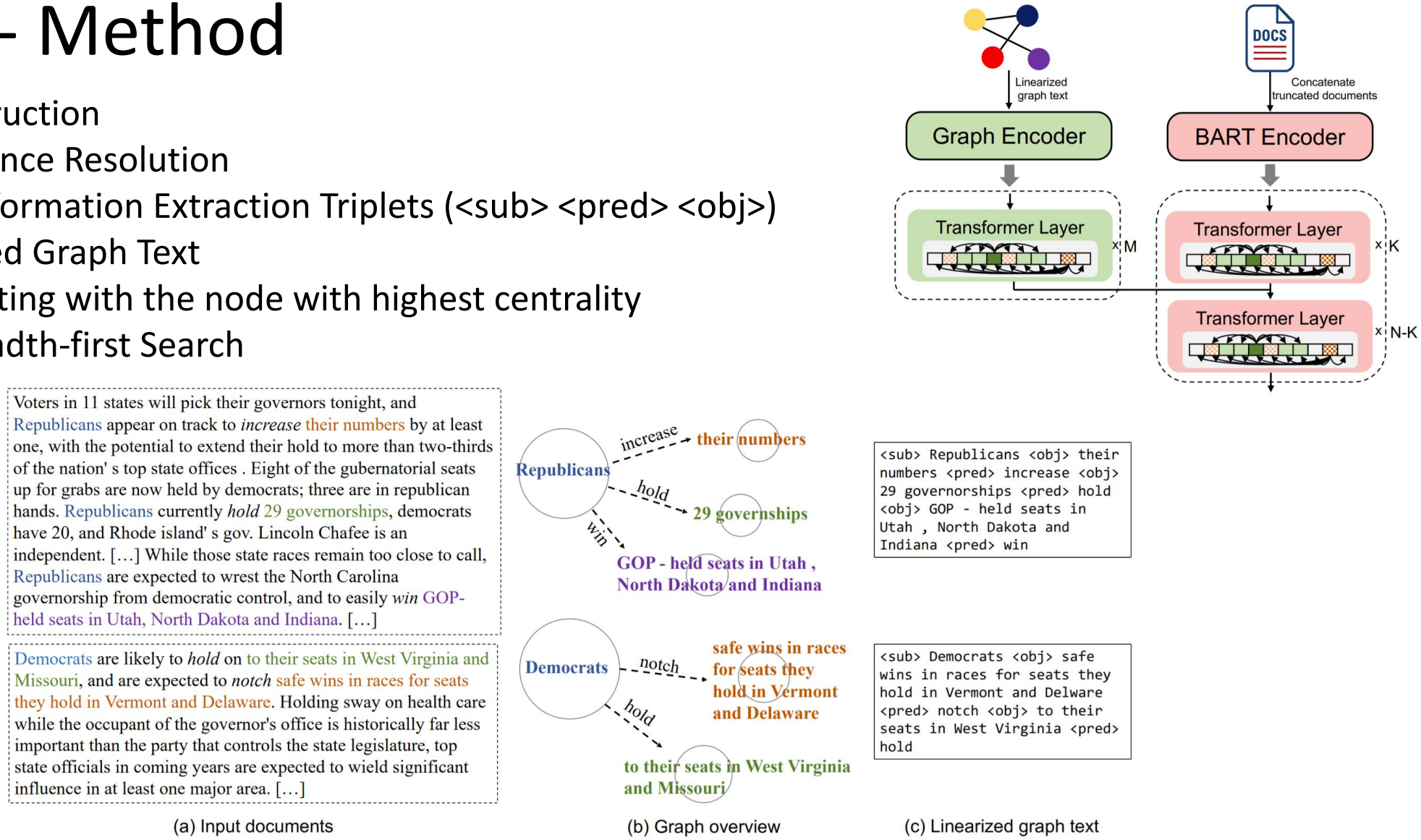


Figure 3: Our graph construction pipeline: (a) Text showing parts of input documents. (b) Overview of graph representation of the information using OIE triplets. (c) Conversion of graph information into text form.

Paper 1 - Experiments

- Datasets
  - Multi-news
  - DUC-2004

Dataset	# pairs	# words (doc)	# sents (docs)	# words (summary)	# sents (summary)	vocab size
Multi-News	44,972/5,622/5,622	2,103.49	82.73	263.66	9.97	666,515
DUC03+04	320	4,636.24	173.15	109.58	2.88	19,734
TAC 2011	176	4,695.70	188.43	99.70	1.00	24,672
CNNDM	287,227/13,368/11,490	810.57	39.78	56.20	3.68	717,951

- Evaluation Metric
  - ROUGE

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)}$$

Paper 1 - Experiments

- Main Result

Model	ROUGE-1	ROUGE-2	ROUGE-L	Average
PREVIOUS WORK				
PG-BRNN (Gehrmann et al., 2018)	43.77	15.38	20.84	26.66
HiMAP (Fabbri et al., 2019)	44.17	16.05	21.38	27.20
Flat Transformer	44.32	15.11	20.50	26.64
Hierarchical Transformer (Liu and Lapata, 2019)	42.36	15.27	22.08	26.57
RoBERTa + Transformer Decoder (Li et al., 2020)	44.26	16.22	22.37	27.62
GraphSum (Li et al., 2020)	45.02	16.69	22.50	28.07
GraphSum + RoBERTa (Li et al., 2020)	45.87	17.56	23.39	28.94
OUR MODELS				
BART-Long	48.54	18.56	23.78	30.29
BART-Long-Graph (500 tokens graph text)	49.03	**19.04**	**24.04**	30.70
BART-Long-Graph (1000 tokens graph text)	**49.24**	18.99	23.97	**30.73**

Table 1: Performance of various models on the Multi-News test set. We report the reproduced results of previous works provided by Li et al. (2020). We report 'summary-level' ROUGE-L scores following Fabbri et al. (2019).

Paper 1 - Experiments

- Transfer Learning on DUC-2004

Model	R-1	R-2	R-SU
EXTRACTIVE METHODS			
MMR (Carbonell and Goldstein, 1998)	30.14	4.55	8.16
LexRank (Erkan and Radev, 2004)	35.56	7.87	11.86
TextRank (Mihalcea and Tarau, 2004)	33.16	6.13	10.16
ABSTRACTIVE METHODS TRAINED ON CNN/DAILY MAIL			
Copy-Transfomer (Gehrmann et al., 2018)	28.54	6.38	7.22
PG-BRNN (Gehrmann et al., 2018)	29.47	6.77	7.56
Hi-MAP (Fabbri et al., 2019)	35.78	8.90	11.43
PG-MMR (Lebanoff et al., 2018)	36.42	9.36	13.23
OUR ABSTRACTIVE METHODS TRAINED ON MULTI-NEWS			
BART-Long (500 text tokens)	33.82	8.09	10.53
BART-Long-Graph (500 text tok. + graph)	34.72	7.97	11.04

Table 2: ROUGE scores on DUC-2004 test-only setup.

Paper 1 - Experiments

- Ablation Study – Input Length

Model	Input Length	R-1	R-2	R-L	Avg.
BART	500	49.22	18.88	23.88	30.66
BART-Long	500	48.54	18.56	23.78	30.29
BART-Long	1000	49.15	19.50	24.47	31.04
BART-Long	1500	48.79	19.14	24.16	30.70
BART-Long	2000	48.96	19.34	24.37	30.89

Table 4: Performance of BART models at various input lengths on the Multi-News dataset.

Paper 1 - Experiments

- Ablation Study – Context window size

Window Size	R-1	R-2	R-L	Avg.
32	48.39	18.75	23.68	30.27
64	48.93	19.28	24.25	30.82
128	48.96	19.34	24.37	30.89
256	49.47	19.94	24.82	31.41
512	49.43	19.86	24.77	31.35

Table 5: Performance of BART-Long model at various attention window sizes. We use Multi-News dataset with 1000 text tokens as input in this comparison to consider longer context window sizes.

Paper 1 - Experiments

- Ablation Study – Target Graph

Target Graph	R-1	R-2	R-L	Avg.
0 %	49.03	19.04	24.04	30.70
25 %	49.99	20.66	24.93	31.86
50 %	54.17	27.30	29.79	37.09
75 %	53.70	28.09	30.28	37.36
100 %	61.32	39.15	38.66	46.38

Table 7: Ablation of the performance of BART-Long-Graph model with 500 tokens input graph text over additionally using a varying percentage of target summary graph information.

Paper 1 - Experiments

- Abstractiveness

Longest Common Subsequence ↑

Model	Density	LCS(%)	4-gr(%) → 4-grams
BL (2,000 text tokens)	15.5	68.8	55.0
BL (1,000 text tokens)	13.6	66.8	50.7
BL (500 text tokens)	11.6	62.5	44.4
BL (500 text tok. + graph)	10.0	59.3	41.3
Reference summaries	5.0	45.9	17.9

Table 8: Abstractiveness: Measuring lexical overlap between summaries and their inputs; lower numbers mean higher abstractiveness. Adding graphs increases abstractiveness.

Paper 2

Multi-Granularity Interaction Network for Extractive and Abstractive Multi-Document Summarization

Hanqi Jin, Tianming Wang, Xiaojun Wan
Wangxuan Institute of Computer Technology, Peking University
Center for Data Science, Peking University
The MOE Key Laboratory of Computational Linguistics, Peking University
{jinhanqi,wangtm,wanxiaojun}@pku.edu.cn

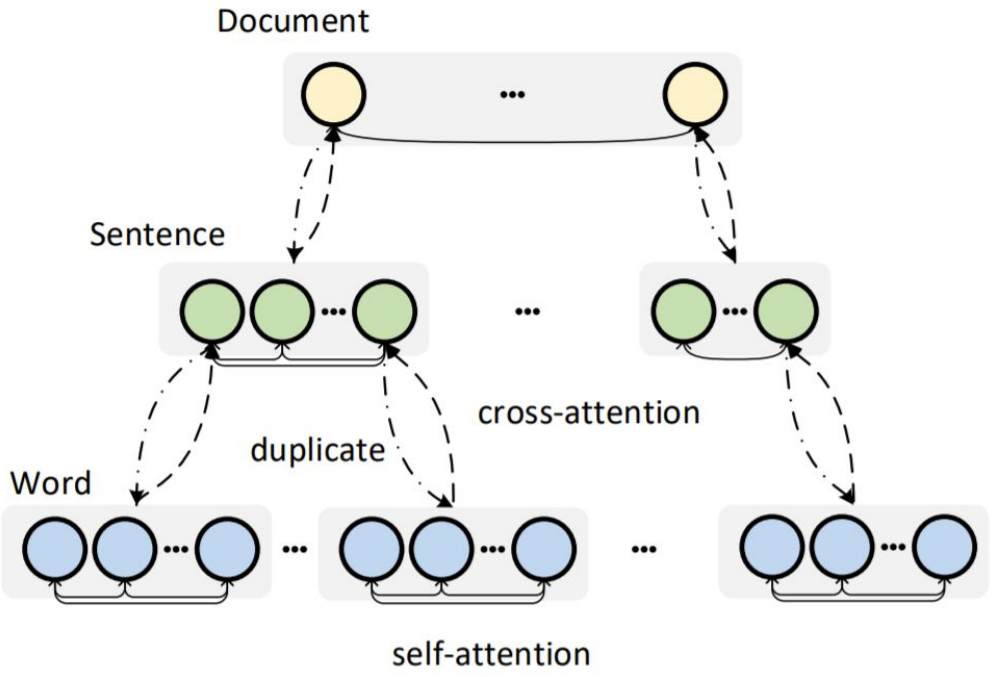
- Multi-Granularity Interaction Network for Extractive and Abstractive Multi-document Summarization (**ACL 2020**)
  - Neural Abstractive & Extractive Summarization
  - Multi-Granularity Attention

Paper 2 - Motivation

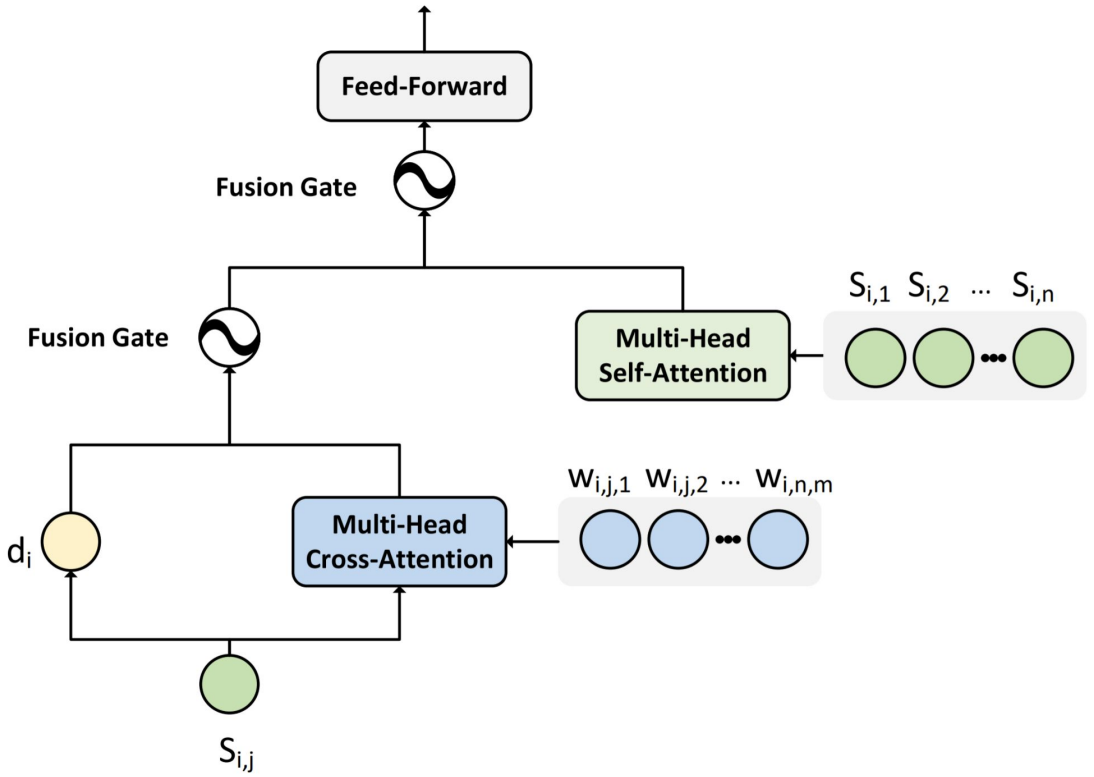
- Limitations of the previous work
  - View multiple documents as a long flat sequence
  - Hierarchical framework mainly focuses on the discourse structure

- Modeling the different granularity of semantic units
  - words, sentences, documents

- Jointly learning abstractive and extractive summarization
  - extractive summarization - sentence-granularity
  - abstractive summarization - word-granularity



Paper 2 - Methods



- Multi-granularity Embeddings
  - Combining embeddings of different granularity together

- Multi-layer Attention

Paper 2 - Methods

$$
\begin{aligned}
f^{l}_{w_{i,j,k}} &= \text{Fusion}\left(\tilde{h}^{l}_{w_{i,j,k}}, \overleftarrow{h}^{l}_{w_{i,j,k}}\right) \\
\tilde{h}^{l}_{w_{i,j,k}} &= \text{MHAtt}\left(h^{l-1}_{w_{i,j,k}}, h^{l-1}_{w_{i,j,*}}\right) \\
\overleftarrow{h}^{l}_{w_{i,j,k}} &= h^{l-1}_{s_{i,j}}
\end{aligned}
$$

- Word-level Encoding

Paper 2 - Methods

$$f^l_{w_{i,j,k}} = \text{Fusion}\left(\tilde{h}^l_{w_{i,j,k}}, \overleftarrow{h}^l_{w_{i,j,k}}\right)$$

$$\tilde{h}^l_{w_{i,j,k}} = \text{MHAtt}\left(h^{l-1}_{w_{i,j,k}}, h^{l-1}_{w_{i,j,*}}\right)$$

$$\overleftarrow{h}^l_{w_{i,j,k}} = h^{l-1}_{s_{i,j}}$$

- Word-level Encoding
  - Word-level Attention

Paper 2 - Methods

$$f^l_{w_{i,j,k}} = \text{Fusion}\left(\tilde{h}^l_{w_{i,j,k}}, \overleftarrow{h}^l_{w_{i,j,k}}\right)$$

$$\tilde{h}^l_{w_{i,j,k}} = \text{MHAtt}\left(h^{l-1}_{w_{i,j,k}}, h^{l-1}_{w_{i,j,*}}\right)$$

$$\overleftarrow{h}^l_{w_{i,j,k}} = h^{l-1}_{s_{i,j}}$$

- Word-level Encoding
  - Word-level Attention
  - Sentence-level Copying

Paper 2 - Methods

$$f^l_{w_{i,j,k}} = \text{Fusion}\left(\tilde{h}^l_{w_{i,j,k}}, \overleftarrow{h}^l_{w_{i,j,k}}\right)$$

$$\tilde{h}^l_{w_{i,j,k}} = \text{MHAtt}\left(h^{l-1}_{w_{i,j,k}}, h^{l-1}_{w_{i,j,*}}\right)$$

$$\overleftarrow{h}^l_{w_{i,j,k}} = h^{l-1}_{s_{i,j}}$$

- Word-level Encoding
  - Word-level Attention
  - Sentence-level Copying
  - Fusion

$$\mathrm{z} = \sigma\left([x; y]W_f + b_f\right)$$

$$\text{Fusion}(x, y) = \mathrm{z}\, x + (1 - \mathrm{z})\, y$$

Paper 2 - Methods

$$\overrightarrow{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{w_{i,j,*}}\right)$$
$$\tilde{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{s_{i,*}}\right)$$
$$\overleftarrow{h}^{l}_{s_{i,j}} = h^{l-1}_{d_i}$$

- Sentence-level Encoding

Paper 2 - Methods

$$\overrightarrow{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{w_{i,j,*}}\right)$$

$$\tilde{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{s_{i,*}}\right)$$

$$\overleftarrow{h}^{l}_{s_{i,j}} = h^{l-1}_{d_i}$$

- Sentence-level Encoding
  - Word-level Attention

Paper 2 - Methods

$$\overrightarrow{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{w_{i,j,*}}\right)$$

$$\tilde{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{s_{i,*}}\right)$$

$$\overleftarrow{h}^{l}_{s_{i,j}} = h^{l-1}_{d_i}$$

- Sentence-level Encoding
  - Word-level Attention
  - Sentence-level Attention

Paper 2 - Methods

$$\overrightarrow{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{w_{i,j,*}}\right)$$

$$\tilde{h}^{l}_{s_{i,j}} = \mathrm{MHAtt}\left(h^{l-1}_{s_{i,j}}, h^{l-1}_{s_{i,*}}\right)$$

$$\overleftarrow{h}^{l}_{s_{i,j}} = h^{l-1}_{d_i}$$

- Sentence-level Encoding
  - Word-level Attention
  - Sentence-level Attention
  - Document-level Copying

Paper 2 - Methods

$$f^l_{s_{i,j}} = \text{Fusion}\left(\text{Fusion}\left(\overrightarrow{h}^l_{s_{i,j}}, \overleftarrow{h}^l_{s_{i,j}}\right), \tilde{h}^l_{s_{i,j}}\right)$$

- Sentence-level Encoding
  - Word-level Attention
  - Sentence-level Attention
  - Document-level Copying
  - Fusion

Paper 2 - Methods

$$
\begin{aligned}
f_{d_i}^{l} &= \text{Fusion}\left(\tilde{h}_{d_i}^{l}, \overrightarrow{h}_{d_i}^{l}\right) \\
\tilde{h}_{d_i}^{l} &= \text{MHAtt}\left(h_{d_i}^{l-1}, h_{d_*}^{l-1}\right) \\
\overrightarrow{h}_{d_i}^{l} &= \text{MHAtt}\left(h_{d_i}^{l-1}, h_{s_{i,*}}^{l-1}\right)
\end{aligned}
$$

- Document-level Encoding

Paper 2 - Methods

$$f_{d_i}^{l} = \text{Fusion}\left(\tilde{h}_{d_i}^{l}, \overrightarrow{h}_{d_i}^{l}\right)$$

$$\tilde{h}_{d_i}^{l} = \text{MHAtt}\left(h_{d_i}^{l-1}, h_{d_*}^{l-1}\right)$$

$$\overrightarrow{h}_{d_i}^{l} = \text{MHAtt}\left(h_{d_i}^{l-1}, h_{s_{i,*}}^{l-1}\right)$$

- Document-level Encoding
  - Sentence-level Attention

Paper 2 - Methods

$$f_{d_i}^{l} = \text{Fusion}\left(\tilde{h}_{d_i}^{l}, \overrightarrow{h}_{d_i}^{l}\right)$$

$$\tilde{h}_{d_i}^{l} = \text{MHAtt}\left(h_{d_i}^{l-1}, h_{d_*}^{l-1}\right)$$

$$\overrightarrow{h}_{d_i}^{l} = \text{MHAtt}\left(h_{d_i}^{l-1}, h_{s_{i,*}}^{l-1}\right)$$

- Document-level Encoding
  - Sentence-level Attention
  - Document-level Attention

Paper 2 - Methods

$$f_{d_i}^{l} = \text{Fusion}\left(\tilde{h}_{d_i}^{l}, \overrightarrow{h}_{d_i}^{l}\right)$$

$$\tilde{h}_{d_i}^{l} = \text{MHAtt}\left(h_{d_i}^{l-1}, h_{d_*}^{l-1}\right)$$

$$\overrightarrow{h}_{d_i}^{l} = \text{MHAtt}\left(h_{d_i}^{l-1}, h_{s_{i,*}}^{l-1}\right)$$

- Document-level Encoding
  - Sentence-level Attention
  - Document-level Attention
  - Fusion

Paper 2 - Methods

$$L_{ext} = -\frac{1}{N}\sum_{n=1}^{N}\Big(y_s^{(n)}\log\tilde{y}_s^{(n)} + \Big(1-y_s^{(n)}\Big)\log\Big(1-\tilde{y}_s^{(n)}\Big)\Big)$$

$$L_{abs} = -\frac{1}{N}\sum_{n=1}^{N}\log p(y_w^{(n)})$$

$$L_{mix} = L_{abs} + \lambda L_{ext}$$

- Extractive Summarization - Using sentence-level embedding
- Abstractive Summarization - Using word-level embedding
- Multi-task Loss

Paper 2 - Experiments

Model	R-1	R-2	R-SU4
Lead-3	39.41	11.77	14.51
LexRank (Erkan and Radev, 2004)	38.27	12.70	13.20
TextRank (Mihalcea and Tarau, 2004)	38.44	13.10	13.50
MMR(Carbonell and Goldstein, 1998)	38.77	11.98	12.91
HIBERT (Zhang et al., 2019)	43.86	14.62	18.34
PGN (See et al., 2017)	41.85	12.91	16.46
CopyTransformer(Gehrmann et al., 2018)	43.57	14.03	17.37
Hi-MAP(Fabbri et al., 2019)	43.47	14.89	17.41
HF(Liu and Lapata, 2019)	43.85	15.60	18.80
MGSum-*ext*	44.75	15.75	19.30
MGSum-*abs*	**46.00**	**16.81**	**20.09**
oracle ext	49.02	29.78	29.19

Table 2: ROUGE F1 evaluation results on the Multi-News test set.

- Automatic Evaluation

Paper 2 - Experiments

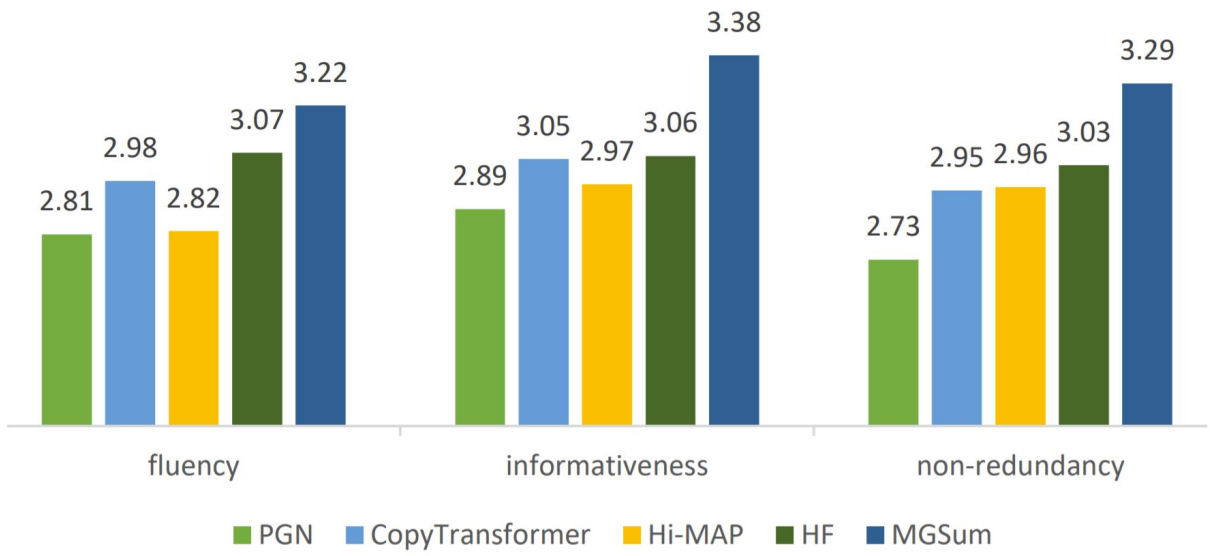


Figure 3: Human evaluation. The compared system summaries are rated on a Likert scale of 1(worst) to 5(best).

- Human Evaluation

Paper 2 - Experiments

Model	R-1	R-2	R-SU4
MGSum-*ext*	45.04	15.98	19.53
only sentence extractor	44.65	15.67	19.27
without doc representation	44.67	15.58	19.15
MGSum-*abs*	46.08	16.92	20.15
only summary generator	45.57	16.32	19.56
without doc representation	45.71	16.62	19.80
without doc&sent representation	44.05	15.31	18.27

Table 3: Results of ablation study on the Multi-News development set.

- Document representation helps

Paper 2 - Experiments

Model	R-1	R-2	R-SU4
MGSum-*ext*	45.04	15.98	19.53
only sentence extractor	44.65	15.67	19.27
without doc representation	44.67	15.58	19.15
MGSum-*abs*	46.08	16.92	20.15
only summary generator	45.57	16.32	19.56
without doc representation	45.71	16.62	19.80
without doc&sent representation	44.05	15.31	18.27

Table 3: Results of ablation study on the Multi-News development set.

- Joint learning helps

More Papers

- Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization (ACL 2019)
- Coarse-to-Fine Query Focused Multi-Document Summarization (EMNLP 2020)
- Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning (EMNLP 2020)
- A Spectral Method for Unsupervised Multi-Document Summarization (EMNLP 2020)

Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization

Sangwoo Cho, Logan Lebanoff, Hassan Foroosh, Fei Liu
Computer Science Department
University of Central Florida, Orlando, FL 32816, USA
{swcho,loganlebanoff}@knight.ucf.edu, {foroosh,feiliu}@cs.ucf.edu

- Determinantal Point Processes (DPP)

- Capsule Network

Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization (ACL 2019)

Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization

- Determinantal Point Processes

$$\mathcal{P}(Y;L) = \frac{\det(L_Y)}{\det(L+I)}$$

$$\sum_{Y \subseteq \mathcal{Y}} \det(L_Y) = \det(L+I).$$

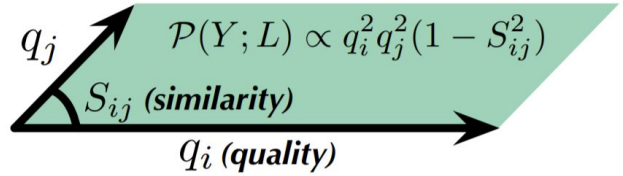


Figure 1: The DPP model specifies the probability of a summary $\mathcal{P}(Y = \{i, j\}; L)$ to be proportional to the squared volume of the space spanned by sentence vectors i and j.

- L_{ij} denotes the similarity between sentence-*i* and sentence-*j*
- Decomposition: $L_{ij} = q_i \cdot S_{ij} \cdot q_j$
- Ideal extractive summary: high quality and low redundancy

$$\begin{aligned}\mathcal{P}(Y = \{i,j\}; L) &\propto \det(L_Y) \\ &= \begin{vmatrix} q_i S_{ii} q_i & q_i S_{ij} q_j \\ q_j S_{ji} q_i & q_j S_{jj} q_j \end{vmatrix} \\ &= q_i^2 \cdot q_j^2 \cdot (1 - S_{ij}^2).\end{aligned}$$

Coarse-to-Fine Query Focused Multi-Document Summarization

Yumo Xu and **Mirella Lapata**
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB
yumo.xu@ed.ac.uk, mlap@inf.ed.ac.uk

- Query Focused Summarization

- Multi-stage Pipeline

Coarse-to-Fine Query Focused Multi-Document Summarization (EMNLP 2020)

Coarse-to-Fine Query Focused Multi-Document Summarization

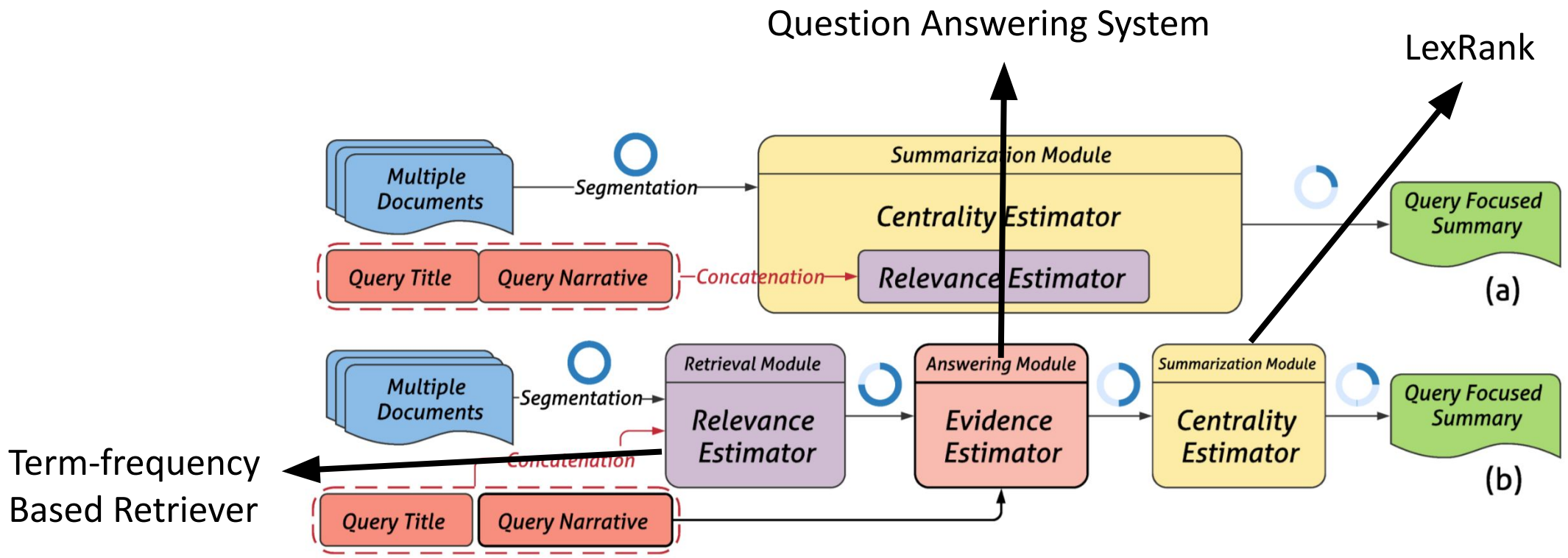


Figure 1: Classic (a) and proposed framework (b) for query-focused summarization. The classic approach involves a relevance estimator nested within a summarization module while our framework takes document clusters as input, and *sequentially* processes them with three individual modules (relevance, evidence, and centrality estimators). The blue circles indicate a coarse-to-fine estimation process from original articles to final summaries where modules gradually discard segments (i.e., sentences or passages). With regard to evidence estimation, we adopt pretrained BERT (Devlin et al., 2019) which is further fine-tuned with distant signals from question answering.

Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning

Yuning Mao[1], **Yanru Qu**[1], **Yiqing Xie**[1], **Xiang Ren**[2], **Jiawei Han**[1]
[1]University of Illinois at Urbana-Champaign, IL, USA
[2]University of Southern California, CA, USA
[1]{yuningm2, yanruqu2, xyiqing2, hanj}@illinois.edu [2]xiangren@usc.edu

- Reinforcement Learning

- Maximal Marginal Relevance

Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning (EMNLP 2020)

Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning

- Maximal Marginal Relevance (MMR)

$$MMR \stackrel{\text{def}}{=} Arg \max_{D_i \in R \setminus S} \left[\lambda (Sim_1(D_i, Q) - (1-\lambda) \max_{D_j \in S} Sim_2(D_i, D_j)) \right]$$

- Combining MMR with neural extractive summarization
  - Hard-cutoff
  - Soft-attention

- Reinforcement Learning Based Extraction
  - Step: extracting one sentence
  - Reward: ROUGE score

A Spectral Method for Unsupervised Multi-Document Summarization

Kexiang Wang[1], **Baobao Chang**[1,2] and **Zhifang Sui**[1,2]
[1]Key Laboratory of Computational Linguistics, Ministry of Education,
School of Electronics Engineering and Computer Science, Peking University, Beijing, China
[2]Peng Cheng Laboratory, Guangdong, China
{wkx,chbb,szf}@pku.edu.cn

- Unsupervised Learning

- Spectral-based Hypothesis

A Spectral Method for Unsupervised Multi-Document Summarization

- Spectral Hypothesis

- Affinity matrix - sentence-wise similarity
  - tf-idf
  - BERT

- Optimization

$$\mathcal{S}^* = \arg\max_{\mathcal{S}\subseteq\mathcal{C}} \Delta\lambda(\mathcal{S}), \quad \text{s.t. } |\mathcal{S}| \leqslant k$$

GIVEN: Affinity matrix $\mathbf{A}$, the matrix representation of document cluster; set $\mathcal{S}$, any summary candidate including some original sentences.

DEFINITION: Spectral impact of $\mathcal{S}$ is the perturbation to dominant eigenvalue of $\mathbf{A}$ when dropping $\mathcal{S}$ from $\mathbf{A}$, i.e. $\Delta\lambda(\mathcal{S}) \triangleq \lambda(\mathbf{A}) - \lambda(\mathbf{A}\backslash\mathcal{S})$.

HYPOTHESIS: Goodness of $\mathcal{S}$ as a summary has a close link with its spectral impact $\Delta\lambda(\mathcal{S})$.

A Spectral Method for Unsupervised Multi-Document Summarization (EMNLP 2020)

Discussion

- What's working?
  - Domain-specific Formulation
    - Saliency
    - Redundancy
    - Coherency
    - ...
  - Deep Learning
    - Pre-trained Language Model
    - Sparse Attention
    - ...
  - Hierarchical Modeling
    - Semantic Granularity
    - Discourse Structure
    - ...

Discussion

- What's working?
  - Domain-specific Formulation
    - Salience
    - Redundancy
    - Coherency
    - ...
  - Deep Learning
    - Pre-trained Language Model
    - Sparse Attention
    - ...
  - Hierarchical Modeling
    - Semantic Granularity
    - Discourse Structure
    - ...

- What's next?
  - More powerful models
  - More and longer documents
  - Revisiting classic algorithms
  - Combining neural and non-neural methods

Drago's Recommended Paper List

1. SUMDocS: Surrounding-aware Unsupervised Multi-Document Summarization
2. Extractive Multi-Document Summarization: A Review of Progress in the Last Decade
3. Query Focused Multi-Document Summarization with Distant Supervision
4. Data Augmentation for Abstractive Query-Focused Multi-Document Summarization
5. Nutri-bullets: Summarizing Health Studies by Composing Segments
6. Nutri-bullets Hybrid: Consensual Multi-document Summarization
7. Abstract Meaning Representation for Multi-Document Summarization
8. Adapting the Neural Encoder-Decoder Framework from Single to Multi-Document Summarization
9. Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization
10. Scoring Sentence Singletons and Pairs for Abstractive Summarization
11. MeanSum : A Neural Model for Unsupervised Multi-Document Abstractive Summarization
12. Learning to Create Sentence Semantic Relation Graphs for Multi-Document Summarization
13. Using Local Knowledge Graph Construction to Scale Seq2Seq Models to Multi-Document Inputs
14. Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning
15. Multi-document Summarization via Deep Learning Techniques: A Survey
16. A Spectral Method for Unsupervised Multi-Document Summarization
17. Hierarchical Transformers for Multi-Document Summarization
18. Recent Advances in Document Summarization (by Jin-ge Yao Xiaojun Wan Jianguo Xiao)
19. WikiAsp: A Dataset for Multi-domain Aspect-based Summarization
20. Affinity-Preserving RandomWalk for Multi-Document Summarization
21. Efficiently Summarizing Text and Graph Encodings of Multi-Document Clusters
22. Sentence Centrality Revisited for Unsupervised Summarization

Questions

- What's the difference between *long* document summarization and *multi*-document summarization? How does the difference affect the algorithms/models?
- How can we automatically collect a large-scale multi-document summarization dataset?
- Given a neural generation model (e.g. BART) with a limited input length (e.g. 1024), how could we use it for much longer input texts?
- How can we combine the supervised methods and unsupervised methods together? Do they complement each other?

References

1. Efficiently Summarizing Text and Graph Encodings of Multi-Document Clusters (NAACL 2021)
2. Multi-Granularity Interaction Network for Extractive and Abstractive Multi-Document Summarization (ACL 2020)
3. Additional papers

- Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization (ACL 2019)
- Query Focused Multi-Document Summarization with Distant Supervision (EMNLP 2020)
- Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning (EMNLP 2020)
- A Spectral Method for Unsupervised Multi-Document Summarization (EMNLP 2020)
- Multi-News: A Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model (ACL 2019)