University of Lethbridge

NSERC CRSNG

Abstractive Unsupervised Multi-Document Summarization using Paraphrastic Sentence Fusion

Mir Tafseer Nayeem, **Tanvir Ahmed Fuad,** Yllias Chali

University of Lethbridge

Lethbridge, Alberta, Canada

What is summarization?

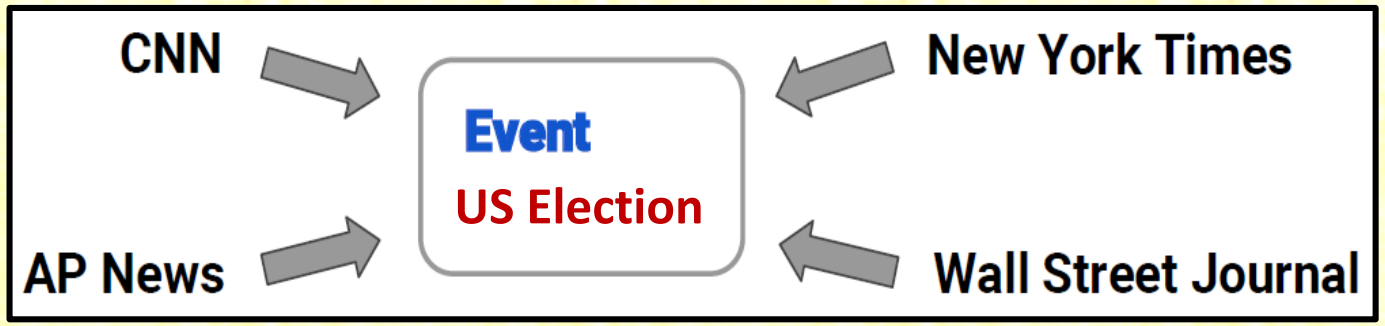
The process of finding the **most relevant informations** in a text and presenting them in a **condensed** form.

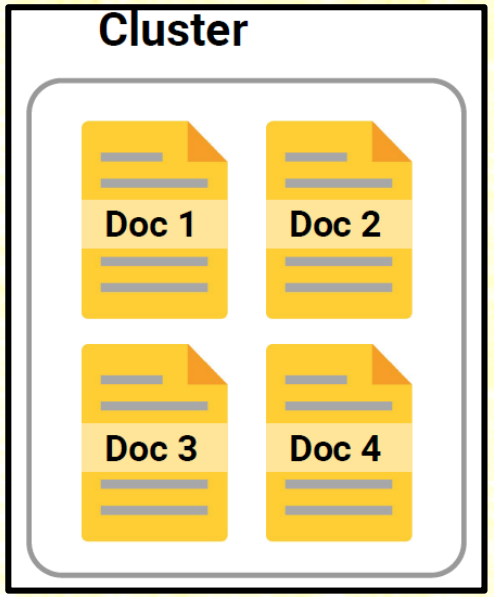
- Single Document Summarization
  - Given a single document produces abstract, outline or headline
- **Multi-Document Summarization**
  - A cluster of related documents about the same topic

- Summaries can be classified as:
  - Extractive
    - Extract important sentences from the original text without any modification.
  - **Abstractive**
    - Abstractive methods rewrite sentences from scratch, involving compression, fusion and paraphrasing.

Why Multi-Document Summarization (MDS)?

- Often times, we want a summary for a whole topic, rather than one document.
  - E.g. different news articles about the same event



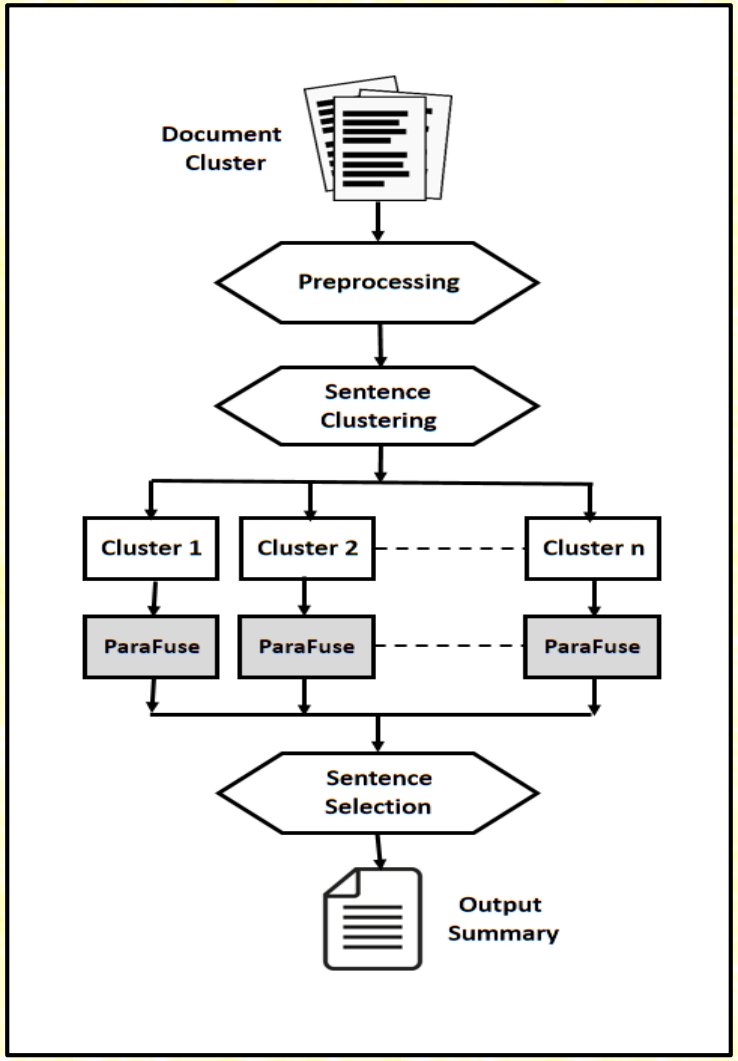


- More challenging, as we need to think about the relationships between documents.

Contributions

- An abstractive sentence generation model is developed which jointly performs sentence fusion and paraphrasing.
- The sentence level model is then applied to design a full abstractive multi-document summarization.
- Different from the recent neural abstractive models, this model is
  - Completely unsupervised
  - Full abstractive
  - Applied to multi-document summarization
  - Domain independent ; tested on news document and user reviews
- An optimal solution is proposed for the classical summary length limit problem in multi-document setting.

Proposed Approach



Presentation Outline

1. **Related Work**
2. Paraphrastic Sentence Fusion
    1. Word Graph Construction
    2. Candidate Ranking
        1. Sentence Embedding
    3. Context Sensitive Lexical Substitution
        1. Substitution Selection
        2. Substitution Ranking
        3. Confidence Score
3. Multi-Document Abstractive Summarization
    1. Sentence Clustering
    2. Abstractive Sentence Selection
    3. Summary Length Limit Problem
4. Experiments
    1. Sentence Level Experiments
    2. Multi-Document Level Experiments

1. Related Work

- Early Works:
  - Word deletion based approaches.
    - Clarke and Lapata 2006, 2008
  - Graph based approaches.
    - Filippova 2010, Boudin and Morin 2013
- Recent Works:
  - Attention based encoder-decoder neural network.
    - Bahdanau 2015, Luong 2015, Cheng and Lapata 2016
  - Seq2seq based learning approaches.
    - Rush 2015
  - Multi-Document based approaches.
    - Yasunaga 2017, Li 2017

Presentation Outline

1. Related Work
2. **Paraphrastic Sentence Fusion**
    1. Word Graph Construction
    2. Candidate Ranking
        1. Sentence Embedding
    3. Context Sensitive Lexical Substitution
        1. Substitution Selection
        2. Substitution Ranking
        3. Confidence Score
3. Multi-Document Abstractive Summarization
    1. Sentence Clustering
    2. Abstractive Sentence Selection
    3. Summary Length Limit Problem
4. Experiments
    1. Sentence Level Experiments
    2. Multi-Document Level Experiments

2. Paraphrastic Sentence Fusion

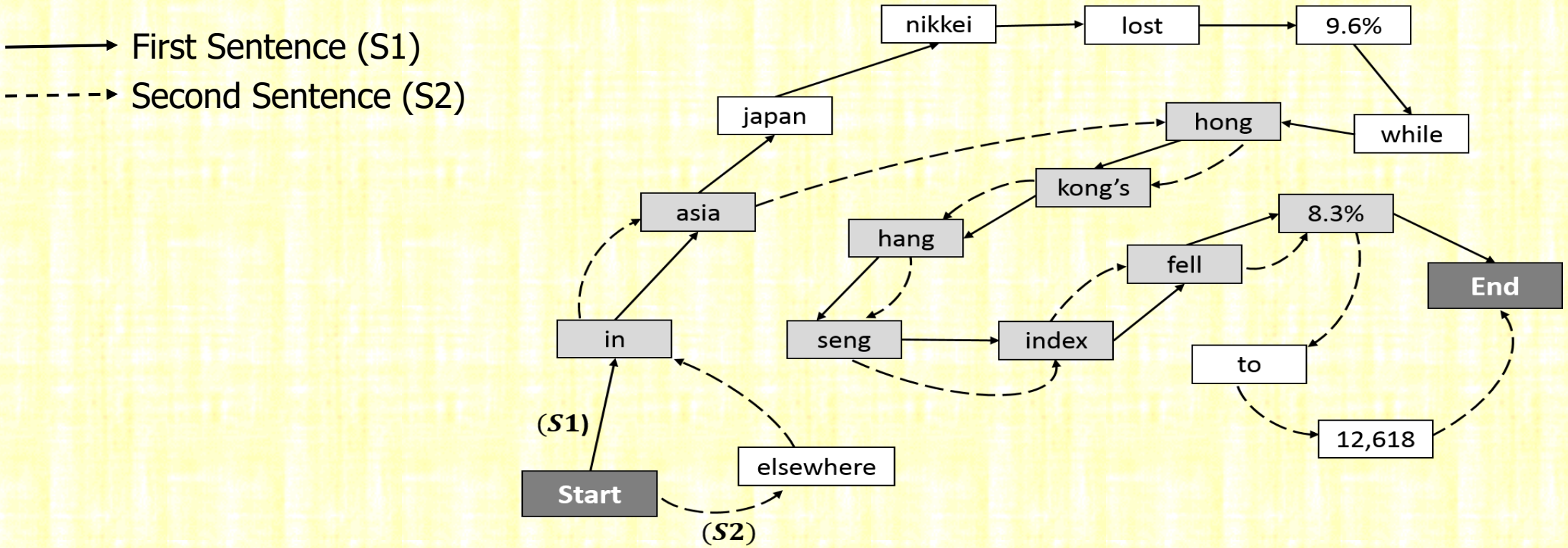
- Most of the previous works are based on deletion based compression.
- Finding representation for sentence abstraction using **sentence fusion** and **lexical paraphrase.**
- We apply our model to the **multi-document abstractive** text summarization.
- Our model balances **information coverage** and **abstractiveness**.

2.1 Word Graph Construction

- We generate a one sentence representation from a **cluster of related sentences** using the **word-graph** approach (Boudin and Morin, 2013).
- $S = \{S_1, S_2, \ldots . S_n\}$ is a cluster of related sentences. We construct a word-graph $G = (V, E)$ by iteratively adding sentences to it.
- The vertices are the words along with the **parts-of-speech (POS)** tags and **directed edges** are the adjacent words in the sentences.
- Each sentence is connected to **dummy start** and **end nodes** to mark the beginning and ending of the sentences.

2.1 Word Graph Construction

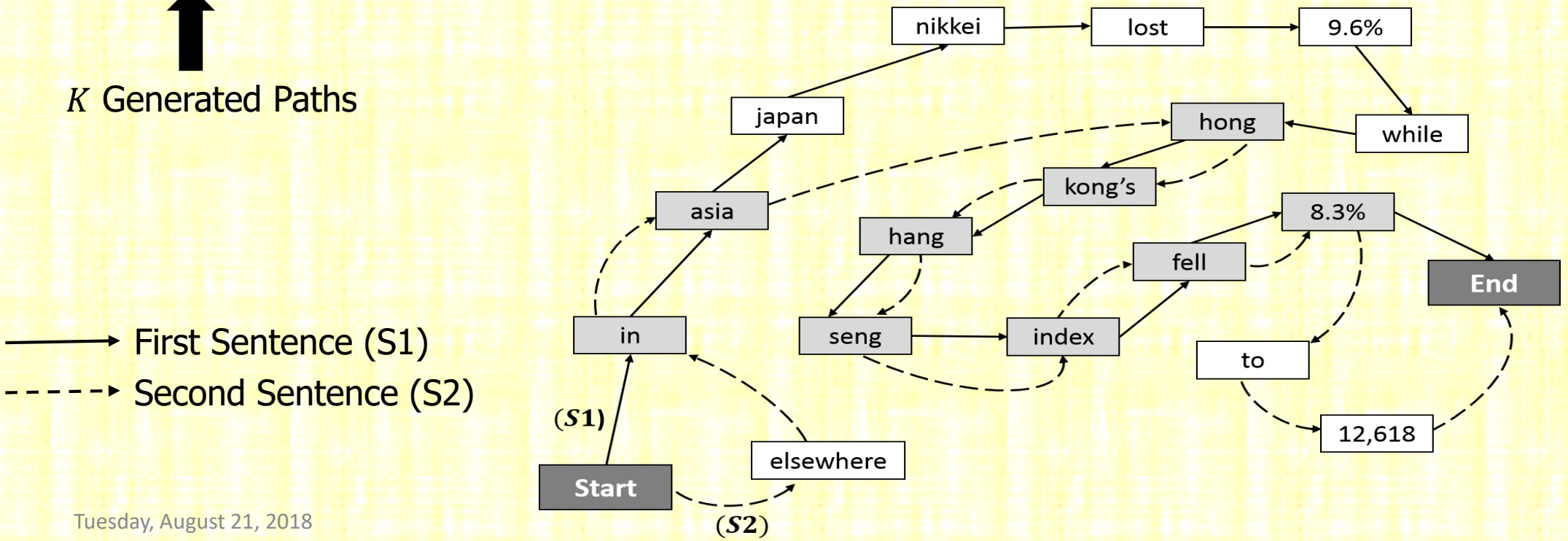
- **S1:** In Asia Japan Nikkei lost 9.6% while Hong Kongs Hang Seng index fell 8.3%.
- **S2:** Elsewhere in Asia Hong Kongs Hang Seng index fell 8.3% to 12,618.



2.1 Word Graph Construction

- **Ex1:** In Asia Hong Kongs Hang Seng index fell 8.3%.
- **Ex2:** Elsewhere in Asia Hong Kongs Hang Seng index fell 8.3%.
- ……
- ……..
- **ExK:** Elsewhere in Asia Japan Nikkei lost 9.6% while Hong Kongs Hang Seng index fell 8.3%.

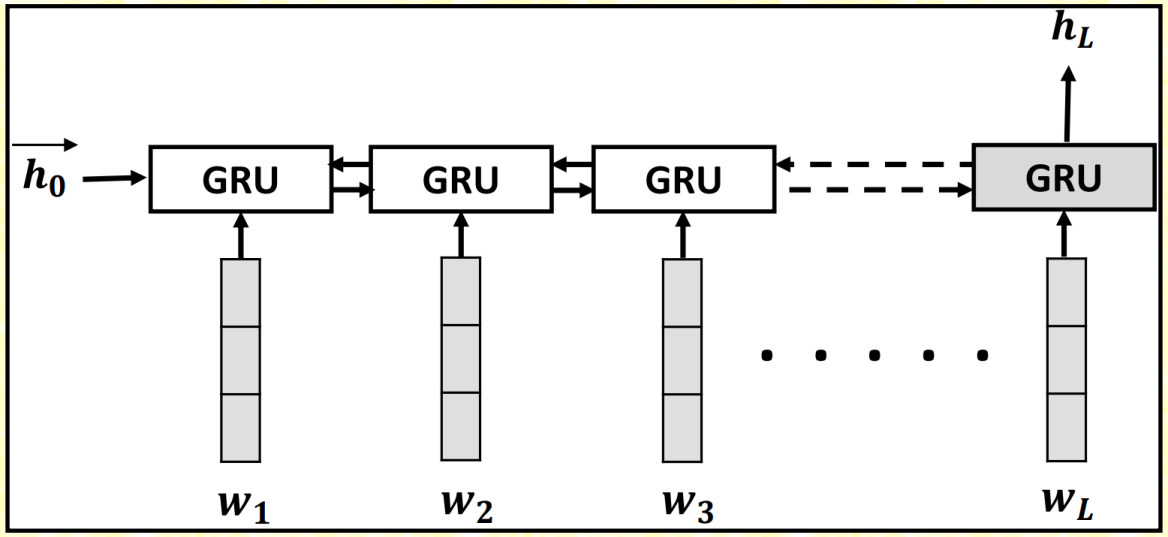
K Generated Paths



2.2 Candidate Ranking

- We rank the sentences using **TextRank** algorithm (Mihalcea and Tarau, 2004).
- An **undirected graph** is constructed where sentences are vertices, and edge weights are the similarity between vertices (sentences).
- Instead of **lexical overlap**, we use the semantic information using sentence embedding.
- After constructing the graph, we can run the **TextRank** algorithm on it by repeatedly applying the updated TextRank rule until convergence.

2.2.1 Sentence Embedding



- Bi-GRU processes the input both from forward and backward direction.
- For each position t, forward and backward states are concatenated into final hidden state $h_t = \overrightarrow{h_t} \oplus \overleftarrow{h_t}$
- Here, $\overrightarrow{h_t} = \boldsymbol{GRU}\left(\overrightarrow{h_{t-1}},\ e(w_t)\right)$ and $\overleftarrow{h_t} = \boldsymbol{GRU}\left(\overleftarrow{h_{t+1}},\ e(w_t)\right)$
- Output sentence embedding $x_i = h_L$ for the sentence S_i

2.2.1 Sentence Embedding

- Sentence, $S = (w_{1,} w_2, \ldots, w_L)$where L is length of the sentence S.

- The sentence is encoded using bi-direcational GRUs.

- For uni-directional case, while reading input:
  - $h_t = \boldsymbol{GRU}\ (h_{t-1},\ e\ (w_t))$
  - Where $h_t \in \mathbb{R}^n$ encodes all content at time t computed from h_{t-1} and $e(w_t)$
  - $e(w_t) \in \mathbb{R}^m$ is the m-dimensional embedding of current word using pre-trained embedding **word2vec**. Here, m=300.

2.3 Context Sensitive Lexical Substitution

- **Target Word Identification for Substitution:** We take only the **nouns** and **verbs** for possible substitution candidates.

- Substitution Selection

- Substitution Ranking

- Confidence Score

2.3.1 Substitution Selection

- **PPDB 2.0** (Pavlick et al., 2015) provides millions of lexical, phrasal and syntactic paraphrases.
- For instance, we can gather lexical substitution set S = {gliding, sailing, diving, travelling} for the target word (**t = flying**) from **PPDB 2.0**.
- We hardcoded the model to select substitutes with the same **POS tag** and that are not a morphological variant (**such as fly, flew, flown**).

2.3.2 Substitution Ranking

- ***word2vecf*** (Levy and Goldberg, 2014) capture functional word similarity (manage → supervise) rather than topical similarity (manage → manager)
- We use the word and context vectors released by (Melamud et al., 2015) which contains 173k words and about 1M syntactic contexts.
- $\boldsymbol{addCos}$ measures the appropriateness of a substitute s from the substitution set S, for the target word t in the set of the target word's context elements

$$C = \{c_1, c_2, \ldots, c_n\} ,$$

$$addCos(s|t,C) = \frac{cos(s,t) + \sum_{c \in C} cos(s,c)}{|C| + 1}$$

- Finally, we select the best substitution s according to maximum $\boldsymbol{addCos}$ scores over 0.7 and replace it with the target word t.

2.3.3 Confidence Score

- When the substitutions are placed, probabilities are assigned to sequence of words in a generated candidate.
- A sequence of m words $\{w_1, w_2, w_3, \dots, w_m\}$. The score **CS** (Confidence Score) assigned to each candidate can be described as:
  - $CS\ (w_1, \dots, w_m) = \frac{1}{1 - Score_{LM}(w_1, \dots, w_m)}$
- In our experiment, a language model is used trained on English Gigaword Corpus

2.3.3 Confidence Score

- K-candidate fusions are ranked and N-best paraphrastic sentence fusions are found which balances **information coverage** and **abstractiveness**.
- Score of a candidate sentence fusion, c is calculated, where $\alpha = 0.5$ to give equal importance,

$$score(c) = \underbrace{\alpha . Rank(c)}_{\textbf{Information Coverage}} + \underbrace{(1-\alpha) . \sum_{V_i=V_{start}}^{V_{end}} addCos(\boldsymbol{V_i}) + CS(N(\boldsymbol{V_i}))}_{\textbf{Abstractiveness}}$$

  - Where, $addCos(\boldsymbol{V_i})$ is the $addCos$ score of the vertex $\boldsymbol{V_i}$ and $N(\boldsymbol{V_i})$ is the neighbors of the vertex $\boldsymbol{V_i}$

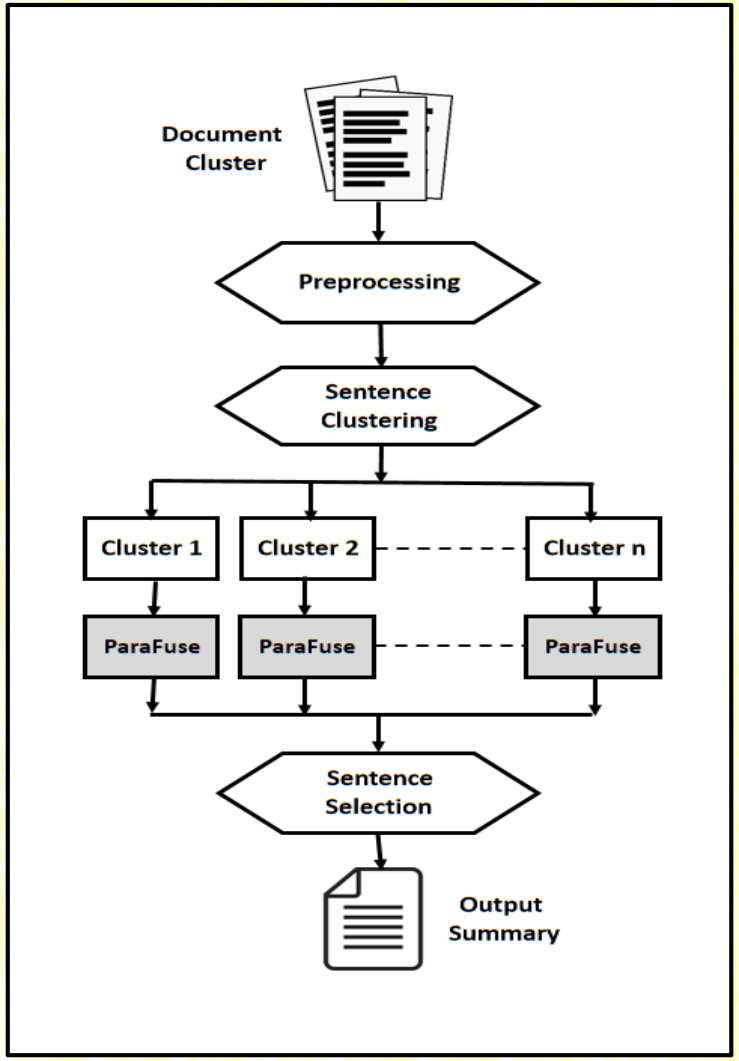
Presentation Outline

1. Related Work
2. Paraphrastic Sentence Fusion
    1. Word Graph Construction
    2. Candidate Ranking
        1. Sentence Embedding
    3. Context Sensitive Lexical Substitution
        1. Substitution Selection
        2. Substitution Ranking
        3. Confidence Score
3. **Multi-Document Abstractive Summarization**
    1. Sentence Clustering
    2. Abstractive Sentence Selection
    3. Summary Length Limit Problem
4. Experiments
    1. Sentence Level Experiments
    2. Multi-Document Level Experiments

3. Multi-Document Abstractive Summarization

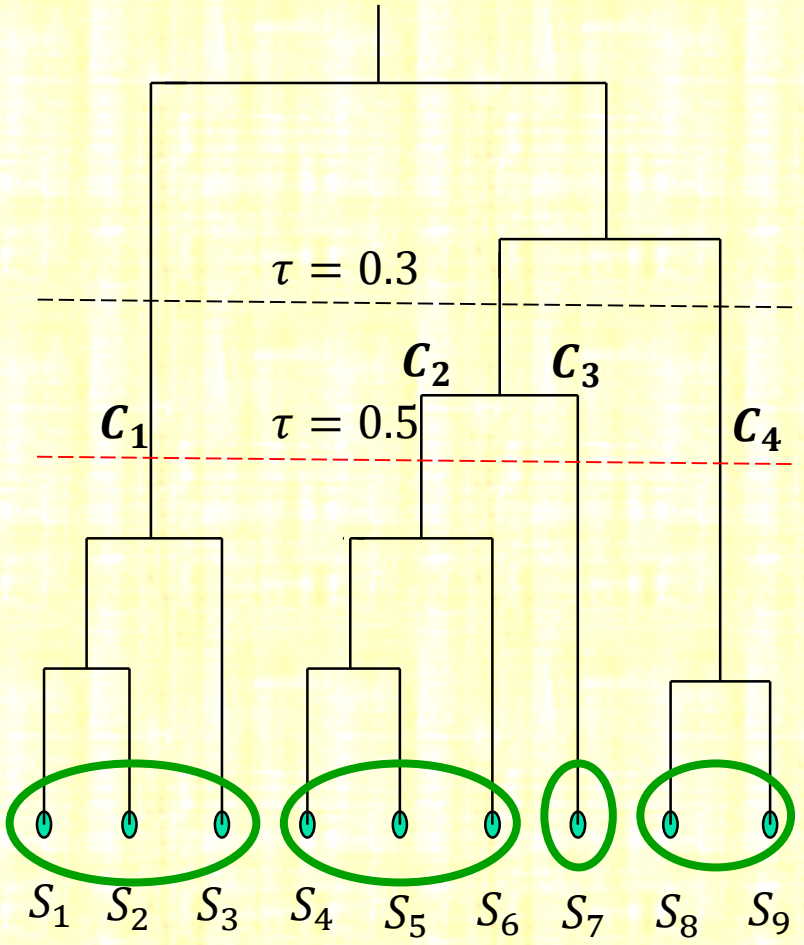
- We apply our paraphrastic fusion model to generate multi-document level summary under a certain length limit.

- Given figure describes our each of the steps involved in multi-document summarization.



3.1. Sentence Clustering

- This step is very important for two main reasons.
  - Selecting at most one sentence from each cluster will **decrease redundancy** from the **summary side**.
  - Selecting sentences from the different set of clusters will increase the **information coverage** from the **document side** as well.
- For grouping similar sentences. We use a **hierarchical agglomerative clustering** (Murtagh and Legendre, 2014) with a **complete linkage** criteria. Similarity threshold (τ = 0.5) was set to stop the process.



3.2. Abstractive Sentence Selection

- We use the **concept-based ILP framework** (Gillick and Favre, 2009) with suitable changes to select the best subset of sentences.
- The system **extracts** sentences that cover **important concepts** while ensuring the **summary length** is within a limit.
- Instead of bigrams we use **keyphrases** as concept.
- We extracted keyphrases using **RAKE** tool (Rose et al., 2010). We assign a weight to each keyphrase using the score returned by RAKE.
- In order to ensure only **one sentence per cluster** we add an extra constraint.

3.2. Abstractive Sentence Selection

Maximize the sum of keyphrase weights

$$max: \sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to: \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.2. Abstractive Sentence Selection

Final score for the candidate sentence

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to : \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.2. Abstractive Sentence Selection

Maximize the summary length

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) - \frac{l_j}{L}) s_j)$$

$$Subject\ to : \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.2. Abstractive Sentence Selection

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to: \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

Summary length under a certain limit

3.2. Abstractive Sentence Selection

Avoiding the repetition of keyphrases

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to : \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.2. Abstractive Sentence Selection

Select at most one sentence from each cluster

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to : \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.2. Abstractive Sentence Selection

Indicates presence of keyphrase

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to : \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.2. Abstractive Sentence Selection

Indicates presence of sentence

$$max : (\sum_i \bar{w}_i k_i + \sum_j (score(s_j) + \frac{l_j}{L}) \cdot s_j)$$

$$Subject\ to : \sum_j l_j s_j \leq L$$

$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

$$\sum_j s_j Occ_{ij} \geq k_i, \quad \forall i$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c$$

$$k_i \in \{0, 1\} \quad \forall i$$

$$s_j \in \{0, 1\} \quad \forall j$$

3.3. Summary Length Limit Problem

- In all the previous research, either truncated at last character or last sentence was removed.
- Recently four methods were proposed to solve this issue:
  - Two of these are based on different decoding procedure
  - Other two are learning based
- Recently proposed methods are also limited to generate a single sentence headline.
- Our model can effectively produce different length variations because of the shortest path strategy.
- In the proposed ILP formulation, the model tries to maximize the total summary length to optimally tackle length limit.

Presentation Outline

1. Related Work
2. Paraphrastic Sentence Fusion
    1. Word Graph Construction
    2. Candidate Ranking
        1. Sentence Embedding
    3. Context Sensitive Lexical Substitution
        1. Substitution Selection
        2. Substitution Ranking
        3. Confidence Score
3. Multi-Document Abstractive Summarization
    1. Sentence Clustering
    2. Abstractive Sentence Selection
    3. Summary Length Limit Problem
4. **Experiments**
    1. Sentence Level Experiments
    2. Multi-Document Level Experiments

4. Experiments

- Sentence Level Experiment.

- Document Level Experiment.

4.1. Sentence Level Experiments

- Dataset:
  - Human generated fusion dataset by McKeown et al 2010
- Evaluation Metric:
  - **BLEU** relies on only exact matching of n-grams.
  - **SARI** which compares **S**ystem output **A**gainst **R**eferences and against the **I**nput sentence. It computes average of n-gram precision and recall of 3 rewrite operations: addition, copying and deletion.
  - **METEOR-E** is an augmented version of **METEOR** using distributed representations.
  - **Compression Ratio** is a measure of how **concise** a compression. A compression ratio of zero implies that the source sentence is fully **uncompressed**.
  - **Copy Rate** how many tokens are copied to the **abstract sentence** from the source sentence without **paraphrasing**. $Copy\ Rate = \frac{|S_{orig} \cap S_{abs}|}{|S_{abs}|}$

4.1. Sentence Level Experiments

- Baseline Systems and Our System:

Input Sentences	Bush, who initially nominated Roberts to replace retiring Justice Sandra Day O'Connor, tapped him to lead the court the day after Rehnquist's death. President Bush initially nominated Roberts in July to succeed retiring Justice Sandra Day O'Connor.
(Filippova, 2010)	president bush initially nominated roberts to replace retiring justice sandra day o'connor .
(Boudin and Morin, 2013)	bush , who initially nominated roberts in july to succeed retiring justice sandra day o'connor , tapped him to lead the court the day after rehnquist 's death .
(Banerjee et al., 2015)	bush , who initially nominated roberts to replace retiring justice sandra day o'connor , tapped him to lead the court the day after rehnquist 's death .
Paraphrastic Fusion (*ours*)	president bush initially **recommended** roberts in july to **substitute** retiring justice sandra day o'connor , tapped him to **run** the court the day after rehnquist 's death .

4.1. Sentence Level Experiments

Model	BLEU	SARI	METEOR-E	Compression Ratio	Copy Rate
(Filippova, 2010)	40.6	34.6	0.31	**0.57**	99.8
(Boudin and Morin, 2013)	**44.0**	37.2	0.36	0.42	99.9
(Banerjee et al., 2015)	42.3	36.5	0.34	0.45	99.8
Paraphrastic Fusion (*ours*)	42.5	**37.4**	**0.43**	0.41	**76.2**

- Our model balances information coverage : **BLUE** and **SARI**.
- Our model completes abstractiveness (**METEOR-E**, **Copy Rate**) instead over compressing (**Compression Ratio**).
- A slightly higher score in **SARI** because of multiple human abstractive rewrites.
- **Copy Rate** clearly indicates that other baseline systems are doing completely deletion based compression.
- Our higher score in **METEOR-E** because of the lexical substitution operation.
- Reasons behind a little bit lower BLEU score:
  - Our model balances between **information coverage** and **abstractiveness**.
  - **BLEU** works well on surface level lexical overlap.

4.2. Multi-Document Level Experiments

- Dataset:
  - DUC 2004 (Length limit = 100 words)
  - Opinosis 1.0 (Length limit = 15 words)
- Evaluation metric:
  - ROUGE-1 (unigram matches)
  - ROUGE-2 (bigram matches)
  - ROUGE-WE (Considering word embeddings to compute semantic similarity)
- We report the **limited length recall** scores for the evaluation metrics.

4.2. Multi-Document Level Experiments

- Results:

Dataset	Models	R-1	R-2	R-WE-1	R-WE-2
DUC 2004	LexRank (Erkan and Radev, 2004)	35.95	7.47	36.91	7.91
	Submodular (Lin and Bilmes, 2011)	39.18	9.35	40.03	9.92
	RegSum (Hong and Nenkova, 2014)	38.57	9.75	39.12	10.33
	ILPSumm (Banerjee et al., 2015)	39.24	11.99	40.21	12.08
	PDG* (Yasunaga et al., 2017)	38.45	9.48	39.07	10.24
	ParaFuse_doc (*ours*)	**40.07**	**12.04**	**42.31**	**12.96**
Opinosis 1.0	TextRank (Mihalcea and Tarau, 2004)	27.56	6.12	28.20	6.45
	Opinosis (Ganesan et al., 2010)	32.35	9.13	33.54	9.41
	Biclique (Muhammad et al., 2016)	33.03	8.96	33.91	9.25
	ParaFuse_doc (*ours*)	**33.86**	**9.74**	**34.46**	**10.09**

Thank You! ☺

Questions?

You can also email us at

mir.nayeem@uleth.ca || t.fuad@uleth.ca