Methods of Sentence Extraction, Abstraction and Ordering for Automatic Text Summarization

Mir Tafseer Nayeem

M.Sc. Candidate

Department of Mathematics and Computer Science

University of Lethbridge

Lethbridge, AB, Canada

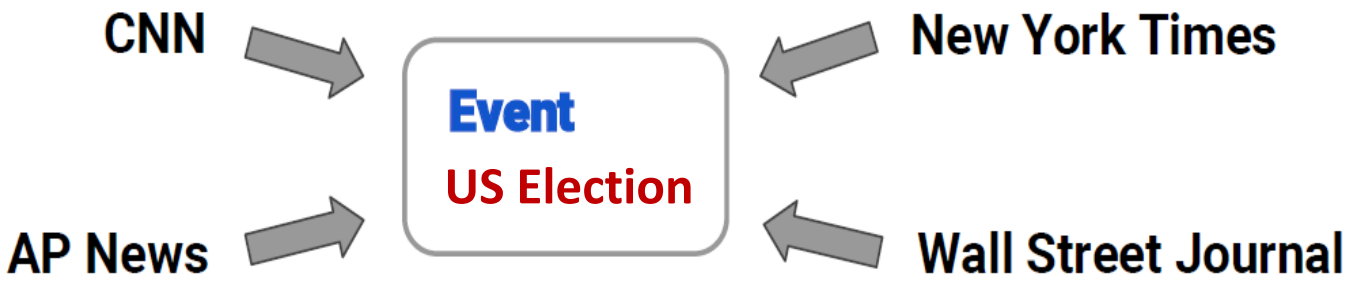
What is summarization?

- The process of **shortening** a text to create a summary with the **major points**.
- **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.

Presentation Outline

- Extractive Multi-Document Summarization
  - Sentence Selection
  - Sentence Ordering
- Abstractive Text Summarization
  - Abstractive Sentence Fusion Generation
  - Multi-Document Abstractive Summarization
- Neural Abstractive Sentence Compression Generation
  - Neural Seq2Seq Paraphrastic Compression model
  - Repetition control using restricted beam search decoding
  - Dealing with out of vocabulary problem
- Neural Abstractive Multi-Document Summarization
  - Optimal Summary Length Limit Problem
- Reader Level Summary Generation
- Future Works

Extractive Multi-Document Summarization

Related Research Works

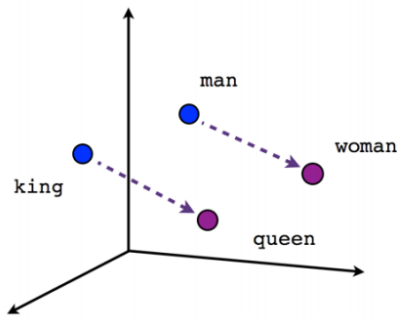
- Early Works:
  - **Graph-based** methods for computing sentence importance.
    - LexRank (Erkan and Radev, 2004) and TextRank (Mihalcea and Tarau, 2004)
  - **Supervised model** for predicting word importance.
    - RegSum system (Hong and Nenkova, 2014)
  - Summarization as a submodular maximization problem (Lin and Bilmes, 2011)
  - All the above systems don't care about the sentence ordering in the output summary.
- Recent Works:
  - **Single document summarization** systems, where sentences are implicitly ordered according to the **sentence position**.
  - Attentional encoder-decoder (Cheng and Lapata, 2016)
  - RNN based sequence classifier (Nallapati et al., 2017)

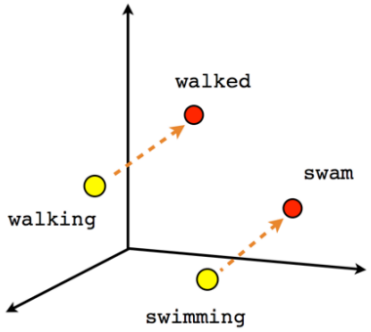
Contributions

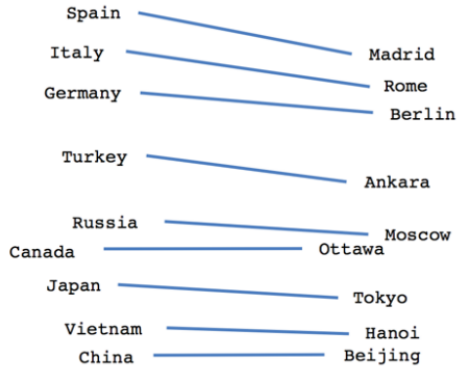
- We implemented an **ILP** (Integer Linear Programming) based sentence selection along with **TextRank** (Mihalcea and Tarau, 2004) scores and **key phrases** for **extractive multi-document** summarization.
- We further model the **coherence** using a greedy algorithm to increase the **readability** of the generated summary.
- We conduct experiments on the Document Understanding Conference (DUC) 2004 datasets using **ROUGE** toolkit.
- Our system achieves significant improvements in terms of **information coverage** and **coherence**.

Sentence Similarity

- We use **Word2Vec** (Mikolov et al., 2013) which embeds words in a continuous vector space where semantically similar words are placed to nearby points to each other.







- It's a popular method used in many natural language processing applications.
- We use the pre-trained word embedding collected from (Mikolov et al., 2013) to represent a sentence.

Sentence Similarity

- Weighted vector sum according to the term-frequency (**TF**) of a word (w) in a sentence (S).
- E is the **word embedding model** (Mikolov et al., 2013) and $idx(w)$ is the index of the word w.

$$S = \sum_{w \in S} TF(w, S) \cdot E[idx(w)]$$

$$Sim(S_i, S_j) = \lambda \cdot NESim(S_i, S_j) + (1 - \lambda) \cdot CosSim(S_i, S_j)$$

Entity Overlap between sentences

Cosine Similarity between sentence vectors

Sentence 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 similarity $Sim(S_i, S_j)$ to form a weighted edge between two sentences.
- After constructing the graph, we can run the **TextRank** algorithm on it by repeatedly applying the following TextRank update rule until convergence.

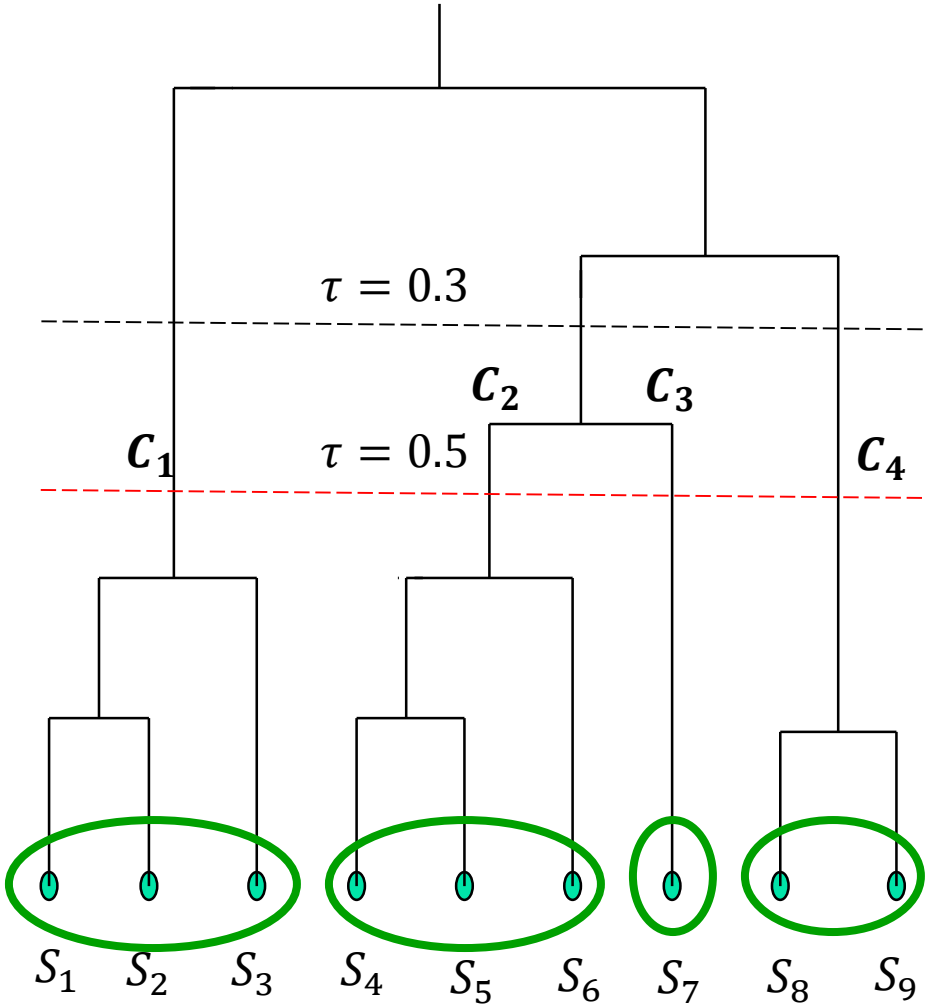
$$Rank(S_i) = (1-d) + d * \sum_{S_j \in N(S_i)} \frac{Sim(S_i, S_j)}{\sum_{S_k \in N(S_j)} Sim(S_j, S_k)} Rank(S_j)$$

- Where $Rank(S_i)$ is the importance score assigned to sentence (S_i), d is the dampening factor which is set to 0.85 as original literature.

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**.
- In computing the clusters, we use the similarity function $Sim(S_i, S_j)$.
- We set a similarity threshold (τ = 0.5) to stop the clustering process.

Sentence Clustering Process



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.

Sentence Selection

$$Maximize: \left(\sum_i \overline{w}_i k_i + \sum_j Rank(S_j) \cdot s_j\right)$$

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

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

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

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

Maximize the sum of keyphrase weights

Sentence Selection

$$Maximize: (\sum_i \overline{w}_i k_i + \sum_j Rank(S_j) \cdot s_j)$$

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

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

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

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

Maximize the sum of sentence rank scores

Sentence Selection

$$Maximize : (\sum_i \overline{w}_i k_i + \sum_j Rank(S_j) \cdot s_j)$$

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

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

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

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

Summary Length under a certain limit

Sentence Selection

$$Maximize : (\sum_i \overline{w}_i k_i + \sum_j Rank(S_j) \cdot s_j)$$

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

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

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

Avoiding the repetition of keyphrases

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

Sentence Selection

$$Maximize : (\sum_i \overline{w}_i k_i + \sum_j Rank(S_j) \cdot s_j)$$

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

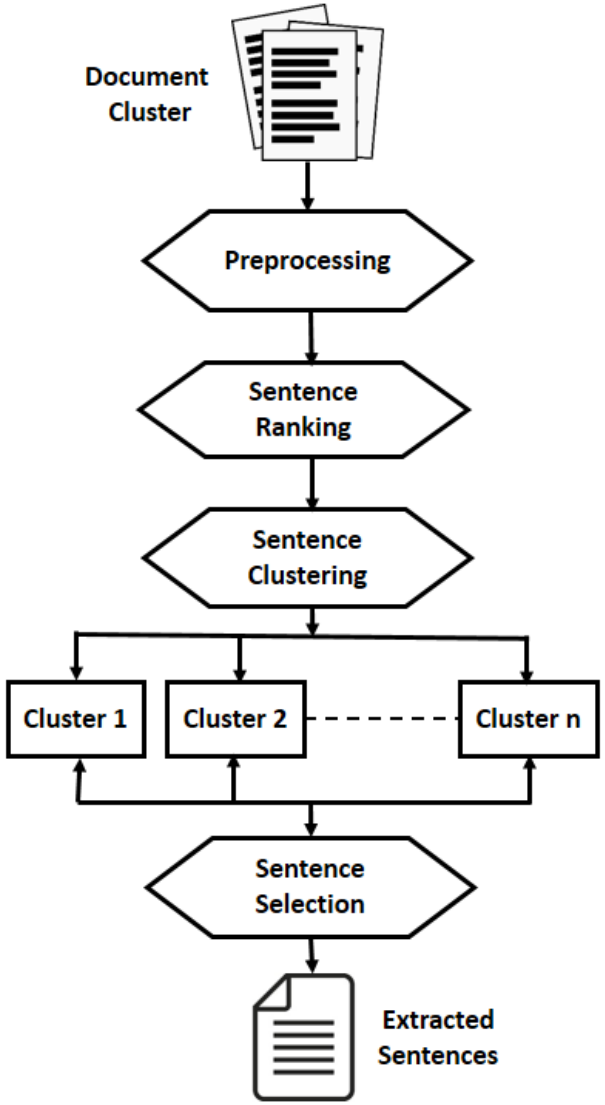
$$s_j Occ_{ij} \leq k_i, \quad \forall i, j$$

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

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

Selects at most one sentence from each cluster

Sentence Extraction Process



Sentence Ordering

- A **wrong order** of sentences convey entirely different idea to the reader of the summary and make it **difficult to understand**.
- For single document, summary can be presented by preserving the **sentence position** in the **original document**.
- **Sentence position** does not provide clue to the sentence arrangement in **multi-document setting**.
- We define **coherence** as the similarity between all adjacent sentences in a document D.

$$Coherence(D) = \frac{\sum_{i=1}^{n-1} Sim(S_i, S_{i+1})}{n-1}$$

Sentence Ordering Algorithm

Algorithm 1: Place a sentence in a document

Procedure `SentencePositioning`(D, S_n)

Data: Input document D which is assumed sorted. New sentence S_n which we will place in the document D.

Result: Return new document D_n after placing the sentence S_n.

$t \leftarrow 1$;
$Coh_{max} \leftarrow 0$;
$D_{tmp} \leftarrow D$;
$l \leftarrow DocLength(D)$;
while $t \leq l + 1$ **do**
  $\Rightarrow$*Place the* S_n *in* t^{th} *position of* D_{tmp} ;
  $Coh_{tmp} \leftarrow Coherence(D_{tmp})$;
  if $Coh_{tmp} > Coh_{max}$ **then**
    $D_n \leftarrow D_{tmp}$;
    $Coh_{max} \leftarrow Coh_{tmp}$;
    $\Rightarrow$ *Remove* S_n *from the* t^{th} *position of the document* D_{tmp} ;
  end
  $t \leftarrow t + 1$;
end
return D_n;

Sample Generated Summary for document set (e.g. d30015t) from DUC-2004 dataset

Summary Generated (After Sentence Extraction)
But U.S. special envoy Richard Holbrooke said the situation in the southern Serbian province was as bad now as two weeks ago. A Western diplomat said up to 120 Yugoslav army armored vehicles, including tanks, have been pulled out. On Sunday, Milosevic met with Russian Foreign Minister Igor Ivanov and Defense Minister Igor Sergeyev, Serbian President Milan Milutinovic and Yugoslavia's top defense officials. To avoid such an attack, Yugoslavia must end the hostilities, withdraw army and security forces, take urgent measures to overcome the humanitarian crisis, ensure that refugees can return home and take part in peace talks, he said.
Summary Generated (After Sentence Ordering)
On Sunday, Milosevic met with Russian Foreign Minister Igor Ivanov and Defense Minister Igor Sergeyev, Serbian President Milan Milutinovic and Yugoslavia's top defense officials. But U.S. special envoy Richard Holbrooke said the situation in the southern Serbian province was as bad now as two weeks ago. A Western diplomat said up to 120 Yugoslav army armored vehicles, including tanks, have been pulled out. To avoid such an attack, Yugoslavia must end the hostilities, withdraw army and security forces, take urgent measures to overcome the humanitarian crisis, ensure that refugees can return home and take part in peace talks, he said.

Evaluation

- Our system **ILPRankSumm** (**ILP** based sentence selection with Text**Rank** for Extractive **Summ**arization)
- Evaluation metric: **ROUGE** Toolkit (Lin,2004)
  - **R-1** (unigram matches)
  - **R-2** (bigram matches)
  - **R-SU4** (skip-bigrams four unigrams in between)
- Dataset : **DUC 2004** (Task-2, Length limit(L) = 100 words)
- We report the **limited length recall** scores for the evaluation metrics.
- ROUGE scores can not determine the **summary coherence**.
- We evaluate summary coherence using (Lapata and Barzilay, 2005) (Barzilay and Lapata, 2008) which output **coherence probabilities** for an ordered set of sentences.

Baseline Systems & Results

- Baseline Systems
  - **LexRank** (Erkan and Radev, 2004)
  - **GreedyKL** (Haghighi and Vanderwende, 2009)
- State-of-the-art Systems
  - **Submodular** (Lin and Bilmes, 2011)
  - **ICSISumm** (Gillick and Favre, 2009)
- The summaries generated by the above extractive summarizers were collected from (Hong et al., 2014)

System	Models	R-1	R-2	R-SU4	Coherence
Baseline	LexRank	35.95	7.47	12.48	0.39
	GreedyKL	37.98	8.53	13.25	0.46
State-of-the-art	Submodular	39.18	9.35	**14.22**	0.51
	ICSISumm	38.41	9.78	13.31	0.44
Proposed System	ILPRankSumm	**39.45**	**10.12**	14.09	**0.68**

Abstractive Text Summarization

Sentence Abstraction : An Overview

- Sentence Abstraction Techniques
  - **Sentence Compression**
  - **Sentence Fusion**
  - **Syntactic Reorganization**
  - **Lexical Paraphrase**

Sentence Compression

- Deletion of unimportant words from the input sentence.
- Used for summarizing a sentence or headline generation.

Sentence Compression

- **Deletion of unimportant words** from the input sentence.
- Used for summarizing a sentence or headline generation.

- **Input Sentence:** "Reporter Jennifer Griffin **has been on the road today** , heading south from Beirut, **and she** joins us by phone from Tyre ."
- **Compressed Sentence:** "Reporter Jennifer Griffin , heading south from Beirut, joins us by phone from Tyre ."

Sentence Fusion

- Involves the merging of two or more sentences into one.

Sentence Fusion

- Involves the merging of two or more sentences into one.
- **Input Sentence #1:** Obama told NBC “I’m frustrated with myself” for unintentionally sending a message that there are “two sets of rules” for paying taxes, “one for prominent people and one for ordinary folks.”
- **Input Sentence #2:** “We can’t send a message to the American people that we have got two sets of rules – one for prominent people and one for ordinary people,” Obama said, defending his administration’s standards.

Sentence Fusion

- Involves the merging of two or more sentences into one.
- **Input Sentence #1:** Obama told NBC "I'm frustrated with myself" for unintentionally sending a message that there are "two sets of rules" for paying taxes, "one for prominent people and one for ordinary folks."
- **Input Sentence #2:** "We can't send a message to the American people that we have got two sets of rules – one for prominent people and one for ordinary people," Obama said, defending his administration's standards.

- **Fused Sentence:** Obama told NBC "I'm frustrated with myself" for unintentionally sending a message to the American people that we have got two sets of rules for paying taxes, one for prominent people and one for ordinary folks.

Syntactic Reorganization

- Helps to make sentence coherent and paraphrase.

Syntactic Reorganization

- Helps to make sentence **coherent and paraphrase**.
- **Input Sentence:** "The cleaning crew vacuums and dusts the office every night."
- **Reorganized Sentence:** "Every night the office is vacuumed and dusted by the cleaning crew."

Lexical Paraphrase

- Replaces complex words with simple words to make the sentence easier to understand.

Lexical Paraphrase

- Replaces **complex words** with **simple words** to make the sentence easier to understand.
- **Input Sentence:** In fact, not many people do think female troops should be **confined** to desk jobs .
- **Paraphrased Sentence:** In fact , not many people do think female troops should be **restricted** to desk jobs .

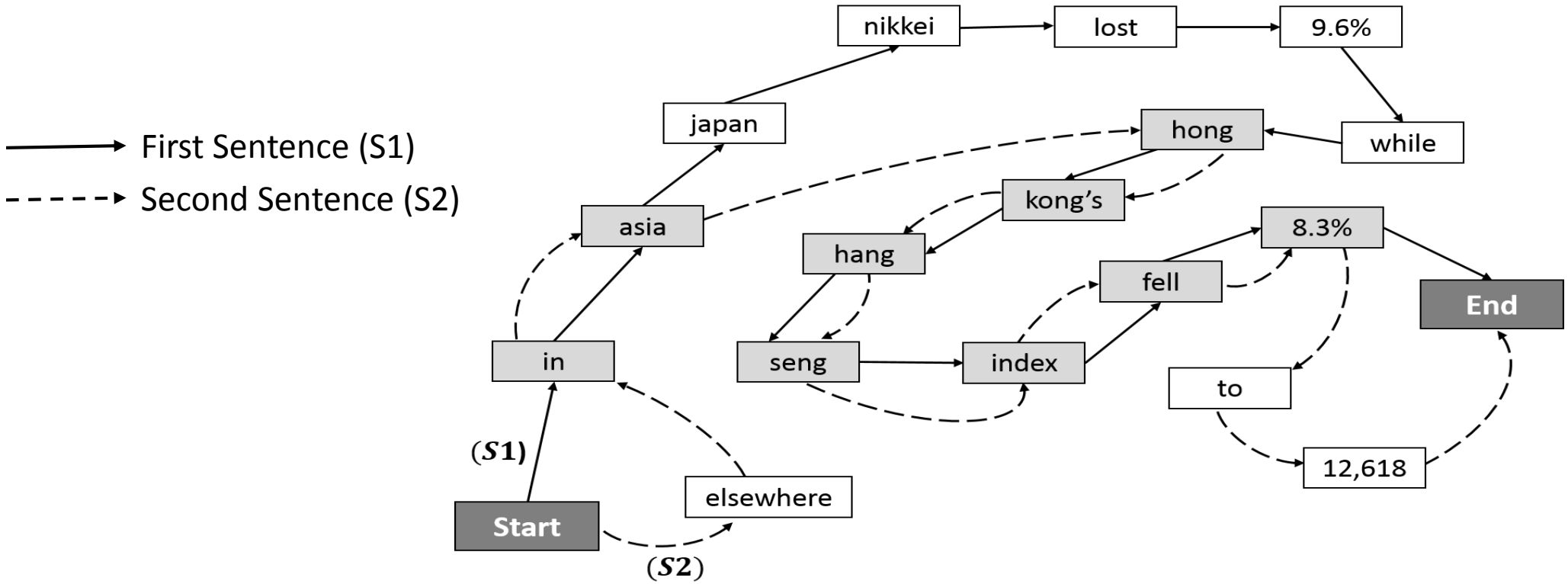
Paraphrastic Sentence Fusion Model

- Jointly models **sentence fusion** and **paraphrasing** using continuous vector representations.
- Try to improve **information coverage** and **grammaticality** of generated sentences.
- We apply our model to the **multi-document abstractive** text summarization.
- Our method brings **significant improvements** over the state of the art systems across different metric.

Paraphrastic Sentence Fusion Model

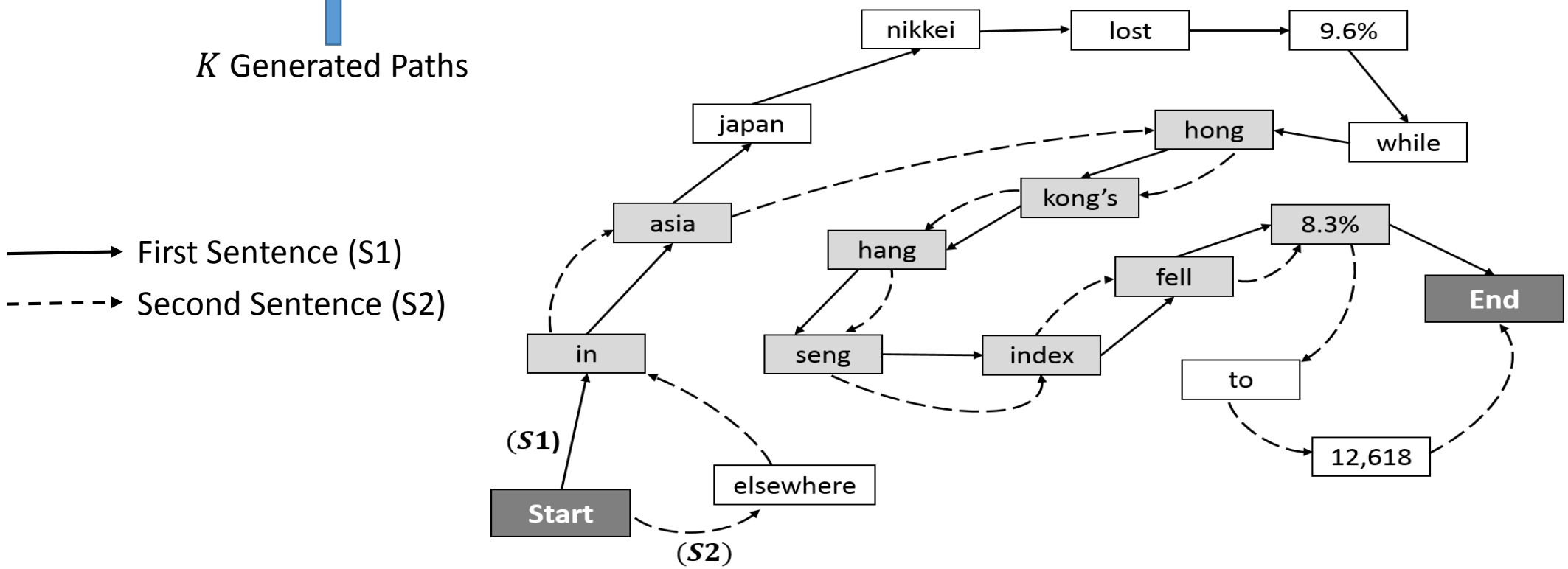
- We generate a one sentence representation from a **cluster of related sentences** using the **word-graph** approach (Boudin and Morin, 2013).
- $S = \{S_1, S_2, \ldots . S_n\}$ is a cluster of related sentences. We construct a word-graph $G = (V, E)$ by iteratively adding sentences to it.
- The vertices 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.

- **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.



- **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



Candidate Re-Ranking

- **Candidate Ranking:** We rank the fused candidates by applying the **sentence ranking** algorithm described earlier.
- **Grammatical Quality:** We compute grammatical quality of a fused sentence candidate using a **3-gram** (trigram) **language model**.

$$GQ(w_1, \ldots, w_m) = \frac{1}{1 - Score_{LM}(w_1, \ldots, w_m)}$$

- Finally, we rank the K candidate fusions and find the ***N*-best sentence fusion** which balances the **grammaticality** and the **informativeness**.

$$score(c) = \alpha \cdot Rank(c) + (1 - \alpha) \cdot GQ(c)$$

Context Sensitive Lexical Substitution

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

Context Sensitive Lexical Substitution

- **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.

Evaluation Metric

- **BLEU** (Papineni et al., 2002) relies on **exact matching of n-grams** and has no concept of **synonymy or paraphrasing**. We used the implementation provided in NLTK considering up to 4-gram matching.
- **SARI** (Xu et al., 2016) a recently proposed metric relies on the availability of **multiple references**. Three rewrite operations: addition, copying, and deletion which correlates well with human references.
- **METEOR-E** (Servan et al., 2016) which uses **word embeddings** along with WordNet synonyms, stemmed tokens and then look-up table paraphrases.

Proposed Evaluation Metrics

- **Compression Ratio** is a measure of how **concise** a compression. A compression ratio of zero implies that the source sentence is fully **uncompressed**.

$$Compression\ Ratio\ (CR) = \frac{\#tok_{del}}{\#tok_{orig}}$$

- **Copy Rate:** how many tokens are copied to the **abstract sentence** from the source sentence without **paraphrasing**.

$$Copy\ Rate = \frac{|S_{orig} \cap S_{abs}|}{|S_{abs}|}$$

- **Grammaticality:** We define grammaticality as the **parsing problem**, if the sentence is successfully parsed, then it has **valid grammar**; if not, then it doesn't. we use a **chart parser** to parse a sentence, given a **CFG** (Context-Free Grammar) which is implemented in NLTK Toolkit.

Experimental Results

- For fair evaluation, we also select the **3-best** candidates for the baseline systems that we compare with our model.
- We conducted experiments on the **human generated sentence fusion** dataset released by (McKeown et al., 2010).
- Consists of **300 English human-produced sentence fusions** rewrites collected via Amazons Mechanical Turk service.

Model	BLEU	SARI	METEOR-E	Compression Ratio	Copy Rate	Gramaticality(%)
Filippova (2010)	40.6	34.6	0.31	**0.57**	99.8	58.2%
Boudin and Morin (2013)	**44.0**	37.2	0.36	0.42	99.9	65.8%
Banerjee (2015)	42.3	36.5	0.34	0.45	99.8	71.4%
Paraphrastic Fusion	42.5	**37.4**	**0.43**	0.41	**76.2**	**73.5%**

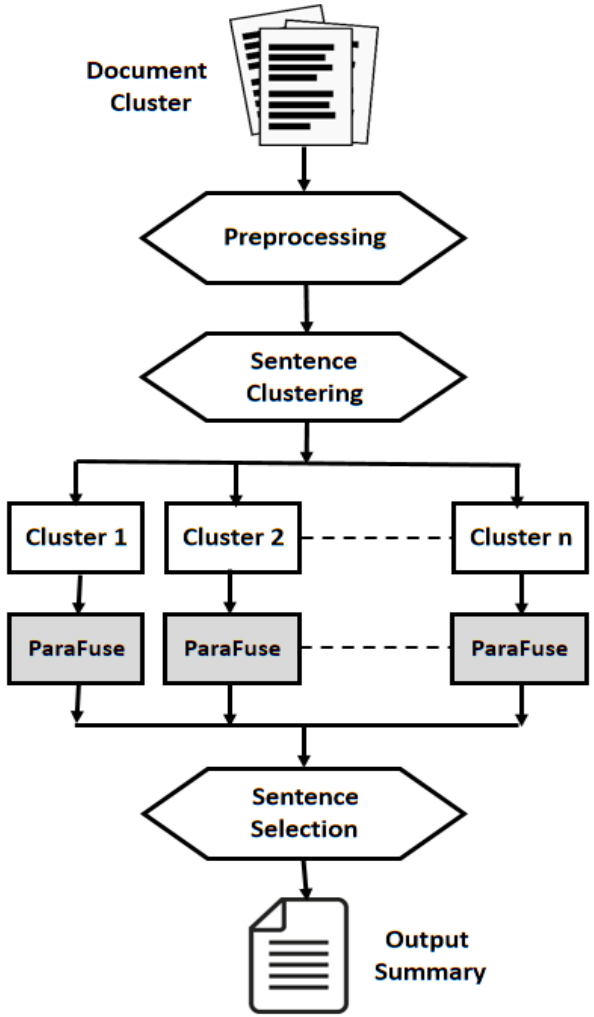
Baselines & Paraphrastic Fusion Model Output

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	president bush initially **recommended** roberts in july to **accomplish** retiring justice sandra day o'connor , tapped him to **run** the court the day after rehnquist 's death .

Document Level Abstractive Summarization

- Our system first takes a set of **related documents** as input according to the same topic.
- We cluster all the sentences in the document using the sentence clustering technique described earlier.
- We use the **concept-based ILP framework** to select the best subset of sentences under certain limit **(L = 100 Words).**
- Finally, we order the extracted sentences using our **greedy sentence ordering** technique.

Proposed document level Paraphrastic Fusion model



Experimental Results

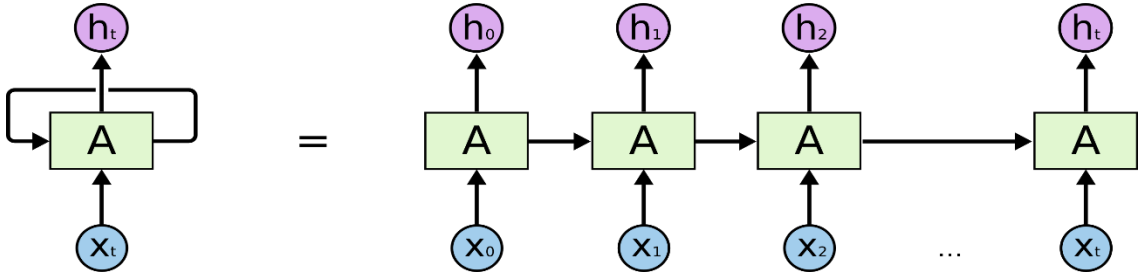
- **ROUGE** (Lin, 2004) scores are **unfairly biased** towards lexical overlap at surface level.
- We also evaluate our system with recently proposed metric **ROUGE-WE** (Ng and Abrecht, 2015), which considers **word embeddings** to compute the semantic similarity of the words.
- We consider the generic multi-document summarization dataset provided at Document Understanding Conference (**DUC 2004**).

System	Models	R-1	R-2	R-WE-1	R-WE-2	Coherence
Baseline	LexRank	35.95	7.47	-	-	0.39
	GreedyKL	37.98	8.53	-	-	0.46
State-of-the-art	Submodular	39.18	9.35	-	-	0.51
	ILPSumm	39.24	11.99	40.31	12.40	0.59
Proposed System	ParaFuse_doc	**40.13**	**12.08**	**42.73**	**13.02**	**0.70**

Neural Abstractive Compression Generation

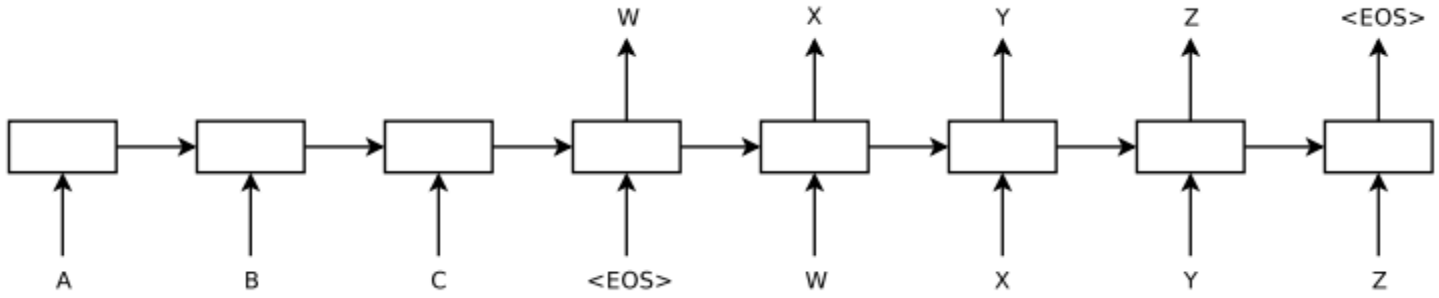
Recurrent Neural Network (RNN)

- In a traditional neural network, we assume that all the inputs and outputs are **independent** on each other.
- **Recurrent neural networks** are generally good for data where there is a relation between previous inputs and the current input in a **sequence** (e.g. Natural Language Texts)
- RNN variants,
  - **LSTM** (Long Short Term Memory)
  - **GRU** (Gated Recurrent Unit)



Neural Machine Translation (NMT)

- Machine translation is actually the task of converting a sequence of words in the source language into a sequence of words in the target language.
- The sequence to sequence networks (or **seq2seq** for short) has received great attention from NLP community to solve the problem of NMT (Sutskever et al., 2014; Bahdanau et al., 2015).
- In **seq2seq**, we can have input and output sequences of different lengths.



Encoder-Decoder Framework

- How does **seq2seq** approach solve that problem of different sequence lengths?
- The answer is: they create a model which consists of two separate recurrent neural networks called **Encoder** and **Decoder** (Cho et al., 2014).
- The encoder turns a source sequence of words into a fixed size feature vector, which is then decoded by a decoder as a target sequence by maximizing the predictive probability.
- Encoder and the decoder are typically implemented via a simple **RNN**, **LSTM** or **GRU**.

Overview of Our Model

- Our neural **Para**phrastic **Comp**ression (**ParaComp**) model based on Neural Machine Translation (NMT).
- **ParaComp** uses neural machine translation to translate from source sentence to an abstractive compression.
- Given a source sentence $\boldsymbol{X} = (x_1, x_2, x_3, \ldots x_N)$ our model learns to predict its paraphrastic compression target $\boldsymbol{Y} = (y_1, y_2, y_3, \ldots x_M)$, where, $M < N$.
- Inferring the target $\boldsymbol{Y}$ given the source $\boldsymbol{X}$ is a typical sequence to sequence learning problem, which can be modeled with **attention-based encoder-decoder models** (Bahdanau et al., 2015; Luong et al., 2015).

Encoder

- The encoder in our case is a bi-directional GRU (**Bi-GRU**) unlike (Luong et al., 2015) which uses uni-directional **LSTM**.
- The **GRU** (Cho et al., 2014) achieves similar performance as **LSTM** but it is fast to train and can improve performance on long sequences.
- Forward **GRU** encodes the source sequence in its original order **left-to-right** and backward **GRU** encodes the source sequence in reverse order, from **right-to-left**.

$$\overrightarrow{h_t} = \mathbf{GRU}(\overrightarrow{h_{t-1}}, e(x_t))$$

$$\overleftarrow{h_t} = \mathbf{GRU}(\overleftarrow{h_{t+1}}, e(x_t))$$

$$h_t = \overrightarrow{h_t} \oplus \overleftarrow{h_t}$$

Decoder

- The decoder uses a simple **GRU** with attention to generate one word y_{t+1} at a time in the paraphrastic compression target sentence $\boldsymbol{Y}$.
- Abstractive sentence generation is conditioned on all previously generated words $y_{1:t}$ and a context vector c_t, which encodes the source sentence:

$$P(\mathbf{Y}|\mathbf{X}) = \prod_{t=1}^{M} P(y_t|y_{1:t-1}, \mathbf{X})$$

$$P(y_{t+1}|y_{1:t}, \mathbf{X}) = softmax\left(g(\mathbf{h}_t^T, \mathbf{c}_t)\right)$$

Attention Mechanism

- While translating a source input sentence, we generally pay more **attention** or concentration to the **relevant words**.
- We allow the decoder to attend the different parts of the source sentence at each time step of the output generation.

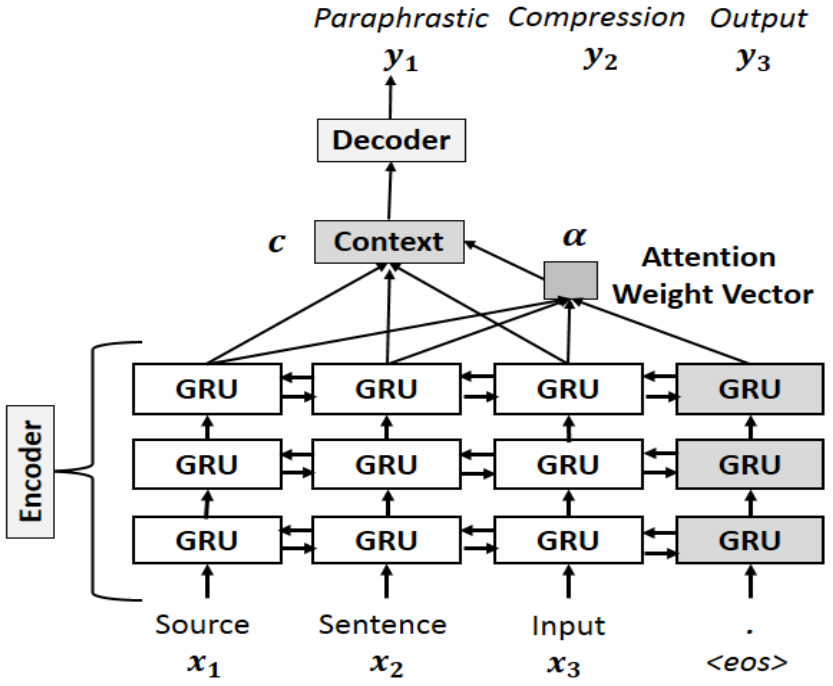
$$\mathbf{c}_t = \sum_{i=1}^{N} \alpha_{ti} \mathbf{h}_{3,i}^{S}$$

$$\alpha_{\mathbf{ti}} = \frac{\exp(\mathbf{h}_t^T \cdot \mathbf{h}_{3,i}^{S})}{\sum_i \exp(\mathbf{h}_t^T \cdot \mathbf{h}_{3,i}^{S})}$$

- Where, c_t is a context vector and α_{ti} denotes the strength of attention of the t-th word in the target language sentence to the i-th word in the source sentence.

Encoder Modification #1

- One important modification we can do to the **bi-GRUs** following (Luong et al., 2015) is stacking multiple layers on top of each other (**stacked GRUs**)



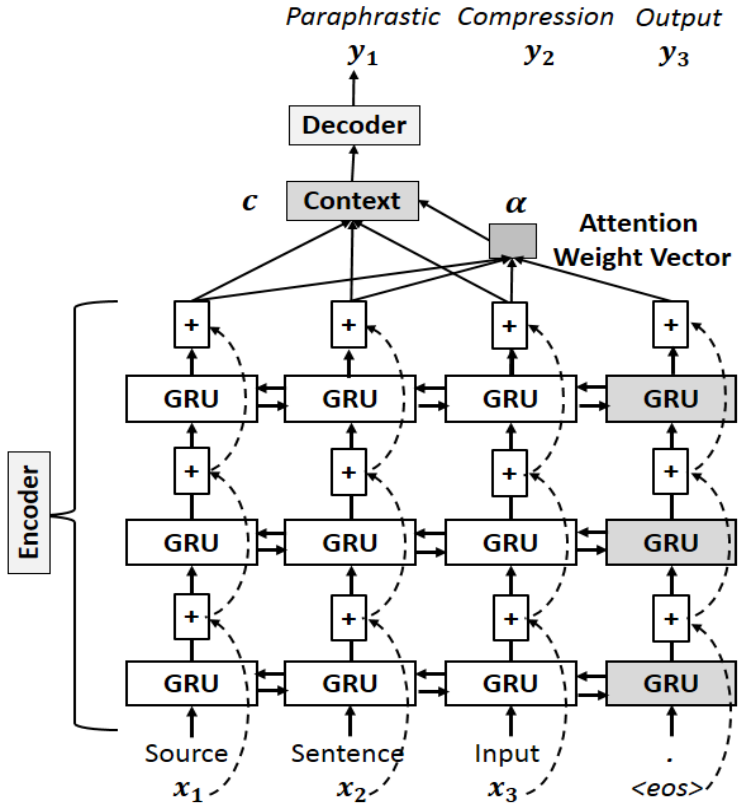
$$h_{1,t} = \mathbf{BiGRU_1}(e(x_t), h_{1,t-1})$$

$$h_{2,t} = \mathbf{BiGRU_2}(h_{1,t}, h_{2,t-1})$$

$$h_{3,t} = \mathbf{BiGRU_3}(h_{2,t}, h_{3,t-1})$$

Encoder Modification #2

- Stacking **RNNs** suffer from the vanishing gradient problem in the vertical direction from the output layer (**GRU3**) to the layer close to the input (**GRU1**).
- This causes the earlier layers of the network to be under-trained.



$$h_{1,t} = \mathbf{BiGRU_1}(e(x_t), h_{1,t-1}) + e(x_t)$$

$$h_{2,t} = \mathbf{BiGRU_2}(h_{1,t}, h_{2,t-1}) + h_{1,t}$$

$$h_{3,t} = \mathbf{BiGRU_3}(h_{2,t}, h_{3,t-1}) + h_{2,t}$$

Out of vocabulary problem

- The output word is selected according to the probability distribution over the whole target vocabulary in the **softmax** layer, which is the most **time and capacity consuming** part of the system.
- Most systems keep a fix-sized target vocabulary according to the word frequency. The infrequent words were removed from the vocabulary and were replaced with the symbol **<UNK>.**
- However, it has been observed that the infrequent words are usually **proper nouns or named-entities** that have an impact on the meaning of the overall sentence.

Solutions

- (Gu et al., 2016) introduced **COPYNET** which is an **encoder-decoder** architecture equipped with **copying mechanism**.
- (Gulcehre et al., 2016) also proposes to solve the unknown word related problem in **end-to-end neural network**. When predicting an output word, the model first makes a decision whether to pick a word from target vocabulary or copy from source input.
- **Our Solution:**
  - We use 100 **<UNK>** placeholders to represent out of vocabulary **<OOV>** words.
  - The **placeholders** are working as a **queue** and taken from the model vocabulary's last 100 places.
  - During generation we copy the unknown words from the input sentence to the placeholders according to **their position** of appearance in the source sentence.

Sequence Repetition Problem

- **Repetition** is a common problem in **seq2seq** encoder decoder model (Tu et al., 2016; Sankaran et al., 2016).

Output#1: It is not appropriate appropriate to insist upon a Syrian withdrawal.

Output#2: Lebanese parliamentary sessions parliamentary session have to be open to the public.

Output#3: This has been ruled has been out .

- **Recent Solutions:**
  - (Suzuki and Nagata, 2017) jointly estimate the **upper bound frequency** of each target vocabulary in the encoder and control the output words.
  - (See et al., 2017) maintain a **coverage vector**, which is the sum of attention distributions over all previous decoder time steps.

Proposed Solution

- Our main goal is to reduce the **complexity in the decoder**.
- For each beam, we keep track of all the previously generated tokens at the t^{th} time step of the decoder in a separate variable called $V_{history}\ (t)$.
- While generating the t^{th} word our model look into the $V_{history}\ (t-1)$ for immediate uni-gram repetition, $V_{history}\ (t-2)$ for bi-gram repetition and $V_{history}\ (t-3)$ for possible tri-gram repetition.
- We hard code the decoder not to choose these words (or any **morphological variation** of these words) which may cause **redundant repetition**.

Restricted Beam Search Decoding Algorithm

Algorithm 2: Restricted Beam Search Decoding Algorithm

Data: Vocabulary size $|V|$, beam size B, max output length N.

Result: Return K paraphrastic compression variations of a source sentence.

$\Rightarrow$ Computed probabilities of all the words in vocabulary

$\Rightarrow$ Choose the B most likely words and initialize the B hypotheses

while $t \leq N$ **do**

$\Rightarrow$ *For each hypothesis, compute the next conditional probabilities, then have* $B \times |V|$ *candidates with the corresponding probabilities*

$\Rightarrow$ *Use the* $[AttentionScore]$ *to choose B most likely candidates those are not in the* $V_{history}[(t-3):(t-1)]$

end

Paraphrasing in Context

- Our model implicitly learned how to paraphrase and can eventually generate paraphrase from the data itself.
- To ensure **complete paraphrasing** we also impose some explicit edit operations.
- **Pre-Edit Paraphrasing**
  - We use **50K** most frequent words, as our model vocabulary out of almost **300K** unique words from the training set.
  - We create an alignment table of **8K** words, for the words outside vocabulary to the words inside vocabulary using **GloVe** embedding (Pennington et al., 2014) having **Cosine Distance ≥ 0.7** (**e.g. pricey ⇒ expensive, detested ⇒ hated etc**)
- **Post-Edit Paraphrasing**
  - We use the **context sensitive lexical substitution** operation presented earlier to accomplish post-edit paraphrasing.

Dataset

- Previous works take the first sentence of a news document, align it with the headline of that document.
- Headlines are not expected to be grammatical and complete.

Datasets	# of Pairs	Source Length	Target Length	# of Vocab
Compression	5,739	22.17	15.81	9.8K
Paraphrase	14,072	22.74	21.91	31.8K
Abstractive Compression	504,543	25.38	18.00	267K
Text Simplification	141,582	25.68	16.97	40K

- In total, we collected almost 665,936 human-generated training pairs for our model.

Training Details

- We trained our model on an **Nvidia TITAN X GPU** card with **12G RAM**.
- We use **300**-dimensional pre-trained **GloVe** embeddings (Pennington et al., 2014).
- We use **reverse training sequence** (Sutskever et al., 2014) which is a trick that avoids long-distance dependencies in **RNN**.
- We use **Adam** (Kingma and Ba, 2014) to optimize parameters with a **mini-batch of size 80**.
- We followed **scheduled sampling** (Bengio et al., 2015) that dynamically adjust the balance between target feeding and self generation.
- **No dropouts. Beam Search size = 10**.

Our Abstractive Compression Generation Model's Output

Source Sentence	It is the right message, sent while it is still early enough to do something constructive about the disappointing quality of the work so far.
Reference(*Best*)	It is the right message to send to correct the disappointing quality of work so far. **(CR: 0.36)**
Output#1	*this* message is the right message. **(CR: 0.76)**
Output#2	it is the right message, sent while it is still early enough to do something *suitable*. **(CR: 0.44)**
Output#3	it is the right message, sent while it is still early enough to do something *faster* about the work. **(CR: 0.24)**
Output#4	*this* message is the right message, sent while it is still early enough to do something *useful* about the work so far. **(CR: 0.12)**
Output#5	it is the right message, sent while it is still early enough to do something *faster* about the work so far. **(CR: 0.16)**

Experimental Results

- **Baseline Systems**
  - **ILP** (Clarke and Lapata, 2008)
  - **T3** (Cohn and Lapata, 2009)
  - **Seq2Seq** (Filippova et al., 2015)
  - **NAMAS** (Rush et al., 2015)

Model	BLEU	SARI	METEOR-E	Compression Ratio	Copy Rate	Gramaticality(%)
T3	11.1	25.7	0.22	**0.75**	90.6	57.2%
ILP	**54.7**	38.1	0.35	0.29	99.5	59.8%
Seq2Seq	53.8	35.5	0.31	0.39	99.7	56.4%
NAMAS	38.7	36.6	0.32	0.24	99.8	49.3%
ParaComp_sent	49.2	**39.3**	**0.41**	0.47	**71.3**	**70.4%**

Neural Abstractive Multi-Document Summarization

Document Level Neural Paraphrastic Compression Model

- We use our proposed sentence extraction technique to extract the **important** and **no-redundant** sentences.
- We put a bigger length limit (L = **200 words**) in our **ILP** formulation for sentence extraction, as our paraphrastic compression model will further compress the extracted sentences.
- We then order the sentences using our **greedy sentence ordering** algorithm.
- For each extracted sentences, we generate **5-best** paraphrastic compressions using our ParaComp model (K = **5**).
- We compute **grammatical quality** of a generated paraphrastic compression sentence candidate.
- We use a **ILP** formulation to select best subset of paraphrastic compressions for each **extracted sentences**.

Fixed Summary Length Problem

- One of the essential properties of the text summarization systems is the ability to generate a summary with a **fixed length**.
- **DUC 2004**, Task-2 (Multi-Document): **Length limit = 100 words**.
- It is highly unlikely, the system generated summary ends at 100th word.
- This creates a confusion whether to include the last candidate sentence in the summary or not.

Example

Sentence to be included ? : In honduras, at least 231 deaths have been blamed on mitch, bringing the storm's death toll in the region to 357, the national emergency commission said saturday.

Sentence Length = 30 Words

Hurricane Mitch killed an estimated 9,000 people throughout Central America in a disaster of such proportions that relief agencies have been overwhelmed. ----------

---

Jerry Jarrell, the weather center director, said Mitch was the strongest hurricane to strike the Caribbean since 1988, when Gilbert killed more than 300 people. "Mitch is closing in," said Monterrey Cardenas, mayor of Utila, an island 20 miles (32 kilometers) off the Honduran coast.

87th Word

Previous Solutions: Case#1 (Hong et al., 2014)

Sentence to be included ? : In honduras, at least 231 deaths have been blamed on mitch, bringing the storm's death toll in the region to 357, the national emergency commission said saturday.

Sentence Length = 30 Words

Hurricane Mitch killed an estimated 9,000 people throughout Central America in a disaster of such proportions that relief agencies have been overwhelmed. ----------

---

Jerry Jarrell, the weather center director, said Mitch was the strongest hurricane to strike the Caribbean since 1988, when Gilbert killed more than 300 people. "Mitch is closing in," said Monterrey Cardenas, mayor of Utila, an island 20 miles (32 kilometers) off the Honduran coast. In Honduras, at least 231 deaths have been blamed on mitch, bringing the

87th Word

Previous Solutions: Case#2 (Hong et al., 2014)

Sentence to be included ? : In honduras, at least 231 deaths have been blamed on mitch, bringing the storm's death toll in the region to 357, the national emergency commission said saturday.

Sentence Length = 30 Words

Hurricane Mitch killed an estimated 9,000 people throughout Central America in a disaster of such proportions that relief agencies have been overwhelmed. ----------

---

Jerry Jarrell, the weather center director, said Mitch was the strongest hurricane to strike the Caribbean since 1988, when Gilbert killed more than 300 people. "Mitch is closing in," said Monterrey Cardenas, mayor of Utila, an island 20 miles (32 kilometers) off the Honduran coast. **<end>**

87th Word

Our Solution

- We tackle this issue in **multi-document setting** by generating multiple paraphrastic compression length variations of a sentence.
- In our ILP formulation for the document level summary generation, we are trying to **maximize the total summary length** to optimally solve the **length limit problem**.
- Under any circumstances, our model can choose a **shorter variation** of a sentence automatically to be included in the summary.

Abstractive Sentence Selection

$$Maximize: \sum_i (GQ(s_i) + Sim(s_{ext_i}, s_i) + \frac{l_i}{\hat{L}}) \cdot s_i$$

$$Subject\ to: \sum_i l_i s_i \leq \hat{L}$$

$$\sum_{i \in g_K} s_i \leq 1, \quad \forall g_K$$

$$GQ(s_i) \geq \alpha$$

$$Sim(s_{ext_i}, s_i) \geq \beta$$

$$Sim(s_{ext_i}, s_i) \leq \gamma$$

Maximize the grammatical quality

Abstractive Sentence Selection

$$Maximize: \sum_i (GQ(s_i) + Sim(s_{ext_i}, s_i) + \frac{l_i}{\hat{L}}) \cdot s_i$$

$$Subject\ to: \sum_i l_i s_i \leq \hat{L}$$

$$\sum_{i \in g_K} s_i \leq 1, \quad \forall g_K$$

$$GQ(s_i) \geq \alpha$$

$$Sim(s_{ext_i}, s_i) \geq \beta$$

$$Sim(s_{ext_i}, s_i) \leq \gamma$$

Defines the margin between near extractive to full abstractive summary.

Abstractive Sentence Selection

$$Maximize: \sum_i (GQ(s_i) + Sim(s_{ext_i}, s_i) + \frac{l_i}{\hat{L}}) \cdot s_i$$

$$Subject\ to: \sum_i l_i s_i \leq \hat{L}$$

$$\sum_{i \in g_K} s_i \leq 1, \quad \forall g_K$$

$$GQ(s_i) \geq \alpha$$

$$Sim(s_{ext_i}, s_i) \geq \beta$$

$$Sim(s_{ext_i}, s_i) \leq \gamma$$

Maximizing the summary length.

Abstractive Sentence Selection

$$Maximize: \sum_i (GQ(s_i) + Sim(s_{ext_i}, s_i) + \frac{l_i}{\hat{L}}) \cdot s_i$$

$$Subject\ to: \sum_i l_i s_i \leq \hat{L}$$

$$\sum_{i \in g_K} s_i \leq 1, \quad \forall g_K$$

$$GQ(s_i) \geq \alpha$$

$$Sim(s_{ext_i}, s_i) \geq \beta$$

$$Sim(s_{ext_i}, s_i) \leq \gamma$$

Length Limit = 100 Words

Abstractive Sentence Selection

$$Maximize: \sum_i (GQ(s_i) + Sim(s_{ext_i}, s_i) + \frac{l_i}{\hat{L}}) \cdot s_i$$

$$Subject\ to: \sum_i l_i s_i \leq \hat{L}$$

$$\sum_{i \in g_K} s_i \leq 1, \quad \forall g_K$$

$$GQ(s_i) \geq \alpha$$

$$Sim(s_{ext_i}, s_i) \geq \beta$$

$$Sim(s_{ext_i}, s_i) \leq \gamma$$

Choose at most one from the 5-best paraphrastic compressions for a single sentence

Abstractive Sentence Selection

$$Maximize: \sum_i (GQ(s_i) + Sim(s_{ext_i}, s_i) + \frac{l_i}{\hat{L}}) \cdot s_i$$

$$Subject\ to: \sum_i l_i s_i \leq \hat{L}$$

$$\sum_{i \in g_K} s_i \leq 1, \quad \forall g_K$$

$$GQ(s_i) \geq \alpha$$

$$Sim(s_{ext_i}, s_i) \geq \beta$$

$$Sim(s_{ext_i}, s_i) \leq \gamma$$

Summary Quality Parameter

Experimental Results

- We randomly select **15 sets of documents** as our validation data for tuning the parameters α, β and γ.
- We set the parameters (α = 0.12 , β = 0.4 and γ = 0.8) based on the validation data for **optimal performance**.
- The rest of the **35 document sets** are used for final evaluation.

System	Models	R-1	R-2	R-WE-1	R-WE-2	Coherence
Baseline	LexRank	35.95	7.47	-	-	0.39
	GreedyKL	37.98	8.53	-	-	0.46
State-of-the-art	Submodular	39.18	9.35	-	-	0.51
	ILPSumm	39.24	11.99	40.31	12.40	0.59
Proposed System	ParaComp_doc	**40.06**	**12.01**	**42.41**	**12.73**	**0.71**

Reader Level Summary Generation

- We for the first time introduced the concept **"Reader Level Summary".**
- The output of the summarization system depends largely on the reader of the summary.
- The readers of a summary can be classified based on,
  - **demographic information** (e.g. age, gender, educational background)
  - **cognitive aspects** (e.g. prior experience, technical skills)
  - **personality traits** (e.g. curiosity, patience, mood and confidence etc)
- Sophisticated systems can be build based on this concept which can extend the document summarization research in a **new level**.

Reader Level Summary Generation

- We simply **further tuned the summary** quality parameters such as α, β and γ based on the reader of the summary (e.g. Children, Non-Native reader, etc.).
- **Non-Native reader:**
  - He/She can expect more **grammatically readable summary**.
  - The original source documents are expected to be grammatical as they are written by the **professional editors**.
  - β and γ represents a window between near extractive to complete **abstractive sentence selection**.
  - α parameter measures **grammatical quality**.
  - In case of **non-native readers**, we set the parameters (α = 0.15 , β = 0.5 and γ = 0.9)

Future Work

- **Jointly extracting** the sentences to maximize the **information coverage** and **readability** while **minimizing redundancy** using a single ILP model.
- Propose a **neural paraphrastic fusion** model using **seq2seq** encoder decoder framework.
- Can modify our **seq2seq** encoder decoder model with recently proposed **hierarchical attention networks** (Yang et al., 2016) to encode full document.
- **Syntactic reorganization** of natural language sentences are extremely difficult. In future, we will try to propose a model for this using **Bidirectional Beam Search** (Sun et al., 2017).
- We will conduct some extensive experiments for our **reader level summary** generation using some **readability metrics**.

Thank You! ☺
Questions?

mir.nayeem@uleth.ca