

Towards Abstractive Multi-Document Summarization Using Submodular Function-Based Framework, Sentence Compression and Merging

Yllias Chali

University of Lethbridge
Lethbridge, Alberta, Canada
chali@cs.uleth.ca

Moin Tanvee

University of Lethbridge
Lethbridge, Alberta, Canada
tanvee@uleth.ca

Mir Tafseer Nayeem

University of Lethbridge
Lethbridge, Alberta, Canada
mir.nayeem@uleth.ca

Abstract

We propose a submodular function-based summarization system which integrates three important measures namely importance, coverage, and non-redundancy to detect the important sentences for the summary. We design monotone and submodular functions which allow us to apply an efficient and scalable greedy algorithm to obtain informative and well-covered summaries. In addition, we integrate two abstraction-based methods namely sentence compression and merging for generating an abstractive sentence set. We design our summarization models for both generic and query-focused summarization. Experimental results on DUC-2004 and DUC-2007 datasets show that our generic and query-focused summarizers have outperformed the state-of-the-art summarization systems in terms of ROUGE-1 and ROUGE-2 recall and F-measure.

1 Introduction

Existing multi-document summarization techniques mainly fall into two categories: extractive and abstractive. Extractive approach selects important source sentences to cover the overall concepts of the document set (Erkan and Radev, 2004; Lin and Bilmes, 2010; Boudin et al., 2015; Parveen and Strube, 2015; Parveen et al., 2015; Cheng and Lapata, 2016; Nallapati et al., 2017). This method is very popular because of its simplicity and speed. But it mostly generates less condensed summaries with redundant information. On the other hand, abstractive summarization is a way of natural language generation and using this approach, it is possible to produce human-like summaries (Rush et al., 2015; Chopra et al., 2016;

Wang and Ling, 2016). It requires deep language understanding. Though this technique is complex and less popular than the extractive approach, it is possible to produce more informative and fluent summary. For generating abstractive summaries, researchers often try to modify the candidate sentences by either shortening and compressing it (Knight and Marcu, 2000; Berg-Kirkpatrick et al., 2011; Filippova et al., 2015) or by merging several sentences which is called sentence fusion (Barzilay and McKeown, 2005; Cheung and Penn, 2014; Bing et al., 2015).

In this paper, we divide the whole task of summarization in two main phases: document shrinking and summarization. In the first phase, we apply sentence compression and merging to produce concise and new candidate sentences for the summary. In the second phase, we represent summarization as a submodular function maximization problem under budgeted constraints. While generating summaries, our system considers three important measures namely importance, coverage, and non-redundancy to ensure summary quality. We design three submodular functions for each these measures. The importance property of the summary considers how much relevant information present in a summary. The coverage measure ranks the sentences based on the fact of how representative they are of the document cluster. The third objective function is designed for measuring non-redundancy of the summaries. This metric assigns a score to a sentence based on how many distinct concepts it contains and how dissimilar it is with the other summary sentences. We design the summarization model for both generic and query-focused summarization. Finally, a modified greedy algorithm is applied which obtains near optimal summaries guaranteed to be within $(1 - 1/\sqrt{e})$ of the optimal solution.

2 Related Work

Most of the research on document summarization are extractive which principally based on two important objectives, namely maximizing the relevance and minimizing the redundancy (Carbonell and Goldstein, 1998; Erkan and Radev, 2004). Besides, formulation of summarization as a maximum coverage problem with knapsack constraint (MCKP) (Takamura and Okumura, 2009; Morita et al., 2011) have been used. Recently, summarization has also been considered as a submodular function maximization (Lin and Bilmes, 2010, 2011; Dasgupta et al., 2013) where greedy algorithms were adopted to achieve near optimal summaries. However, the main drawback of all the extractive approaches is that they can not avoid the inclusion of insignificant information which degrades the summary quality.

On the other hand, the abstractive approach in a multi-document setting aims at generating summaries by deeply understanding the contents of the document set and rewriting the most relevant information in natural language. Two recent abstractive techniques are most commonly used to accomplish the task: sentence compression (Knight and Marcu, 2000) and sentence fusion (Barzilay and McKeown, 2005). In the recent years, sentence compression is jointly used with the extractive system to improve summary quality (Berg-Kirkpatrick et al., 2011; Martins and Smith, 2009). In addition, sentence fusion-based models have also been proposed where sentence fragments from multiple sentences are combined to cover more information in a concise manner (Barzilay and McKeown, 2005; Filippova et al., 2015; Ganesan et al., 2010; Thadani and McKeown, 2013; Cheung and Penn, 2014; Bing et al., 2015).

3 Document Shrinking

In this phase, we used sentence compression and sentence merging to prepare a better and more concise document set before approaching the actual summarization task.

3.1 Sentence Compression

Sentence compression is a technique of shortening sentences which can be used with the extractive system to improve summary quality. Consider the following example sentence as a candidate sentence of the summary:

“According to a newspaper report, a total of 4,299 political opponents died or disappeared during Pinochet’s term.”

In this sentence, we can see the part shown in the italic font is not carrying much significance and can be removed. We removed these sort of insignificant sub-parts of sentences following Berg-Kirkpatrick et al., (2011)’s compression technique.

In addition, we removed the sub-clauses related to the reporting verbs from sentences following (Chali and Uddin, 2016), like in the following example sentence:

Cambodian parties agreed to a Coalition government led by Hun Sen, the official said.

We considered mostly used reporting verbs such as *said*, *told*, *reported*, and *announced* to find out subclause. It is known that the sentence which contains a reporting verb is always the ‘root’ of the dependency tree. Following this rule, we traversed the tree to find out the subclause related to the reporting verb and removed it from the sentence.

3.2 Sentence Merging

Sentence merging is a technique to create a more informative sentence by merging the information from different source sentences. According to Bing et al., (2015), human summary writers usually merge the important facts in different verb phrases (VPs) about the same entity into a single sentence. Based on this assumption, we design a sentence merging technique. While Bing et al., (2015), took phrases as the basic linguistic unit and merge phrases to produce a summary, we take sentences as the basic linguistic units and merge them to generate new sentences for the summary. For example, the following sentences: (1) *Cambodian prime minister Hun Sen has ruled through violence*, (2) *Hun Sen threatened to eliminate opponents* can be merged as (3) *Hun Sen has ruled through violence and threatened to eliminate opponents*. For merging two sentences, we identify the sentences which start with a coreferent subject in order to preserve the gramaticality of the newly generated sentence, which is a key challenge in abstractive summarization.

Our system first applies Stanford Coreference Resolution engine (Lee et al., 2013) on each sentence of a document. From this step, we obtain a set of clusters containing the noun phrases that refer to the same entity in a document. A new sen-

tence is generated from two sentences if they share a coreferent NP as the subject but have different VPs. We picked the sentences closest to each other for merging and produced the new sentences. The natural order of the sentences has thus been preserved.

After this phase, we obtain a cluster of documents containing concise sentences. Now, this document set is the input of our document summarization phase.

4 Document Summarization

We consider text summarization as a budgeted submodular function maximization problem similar to the recent works of (Lin and Bilmes, 2011), but our proposed monotone submodular objective function is significantly different from their work, which is discussed in this section.

4.1 Problem Definition

Suppose U be the finite set of all textual-units (sentence) in the documents. Our task of summarization is to select a subset $S \subseteq U$ that maximizes the submodular function. Since there is a length constraint in standard summarization tasks (e.g., DUC¹ evaluations), we consider the problem as a submodular function maximization with budgeted constraints:

$$\max_{S \subseteq U} \left\{ f(S) : \sum_{i \in S} cost_i \leq B_{max} \right\} \quad (1)$$

where, $cost_i$ is the non-negative cost of selecting the textual-unit i and B_{max} is the budget. The value of B_{max} could be the number of words or bytes in the summary. $f(S)$ is the submodular objective function that scores the summary quality.

4.1.1 Generic Summarization

We design a monotone submodular objective function composed of three important objectives for document summarization. These objectives are responsible for measuring summary’s importance, coverage and non-redundancy property. The proposed objective function is:

$$f(S) = \alpha r(S) + \beta c(S) + \Lambda h(S) \quad (2)$$

where, $r(S)$ measures summary’s importance quality, $c(S)$ measures summary’s coverage quality, $h(S)$ measures summary’s non-redundancy

quality and α , β , and Λ are non-negative trade-off coefficients which can be tuned empirically².

As we know, the linear combination of the submodular functions is submodular (Lin and Bilmes, 2011) and all the proposed subparts of our objective function are submodular, the function $f(S)$ is also submodular.

One of the basic requirements of a good summary is that it should contain the most important information across multiple documents. To model this property, we introduce a new monotone nondecreasing submodular function based on the *atomic concept*. In our definition, atomic concepts are the atomic terms that bear significance in a sentence. Our system, therefore, considers only verbs, named-entities, and adjectives as atomic concepts (excluding the stop words). Our proposed submodular function is:

$$r(S) = \sum_{i=1}^N \frac{1}{pos(S_i)} \Omega_i \cdot \lambda_{S_i} \quad (3)$$

where, $\lambda_{S_i} \in \{0, 1\}$, $\lambda_{S_i} = 1$ if sentence S_i is in the summary, otherwise $\lambda_{S_i} = 0$. Ω_i is the importance score of sentence S_i and $pos(S_i)$ denotes the position of sentence S_i in the document.

We consider the relevance of the summary as the summation of the importance scores of the sentences in it. First, we utilize the Markov random walk model used by (Hong and Nenkova, 2014; Mihalcea and Tarau, 2004) to score each concept from the document set. Then we score every sentence based on the weight of the constituent words in the sentence. We only decrease the weight of the constituent concepts when it appears in multiple sentences in the summary. While sentence similarity-based approaches (Lin and Bilmes, 2011) do not consider the individual word’s importance to model the importance property, our proposed submodular function is based on the atomic concept and this model encourages coverage of most of the important concepts across the documents.

A good summary has the capability to cover most of the important aspects of a document set. To formulate this, we consider a submodular objective function which utilizes the following ‘sentence similarity-based approach’ based on ‘facil-

²The values for the coefficients are 1.0, 1.0 and 5.0 for α , β , and Λ respectively, as found empirically using DUC-2003 development set during the experiments.

¹<http://www-nlpir.nist.gov/projects/duc/index.html>

ity location objective” (Cornuejols et al., 1977).

$$d(S) = \sum_{i \in V} \max_{j \in S} \text{sim}(i, j) \quad (4)$$

where, $\text{sim}(i, j)$ denotes the deep semantic sentence similarity between sentence i and j . For measuring the similarity between sentences, we used the Word2Vec sentence similarity measure (Mikolov et al., 2013). We first remove all the stop words³ which do not add much meaning to the sentence and then run Word2Vec⁴ on the words in both sentences. We calculate the average vector for all words in both sentences and use cosine similarity between vectors to find the semantic similarity between sentences. Finally, following equation (4), a sentence’s eligibility to be included in the summary depends on how similar it is with all the other sentences in the document cluster.

Minimizing redundant information in the summary is handled by the following submodular function:

$$h(S) = \sum_{C_k \in \eta(S)} \sigma(C_k) - \sum_{i, j \in S, i \neq j} \text{sim}(i, j) \quad (5)$$

where, $\text{sim}(i, j)$ is the deep semantic sentence similarity between summary sentence i and j , $\sigma(C_k)$ is the weight of k -th concept term, and $\eta(S)$ is the set of all distinct terms in the summary.

The first part of the function $h(S)$ is based on atomic-concept which scores the summary by measuring the weighted sum of the unique concept terms in the summary. In the second part, we penalize the summary redundancy by measuring semantic similarity among the summary sentences. Finally, our task is to maximize the proposed submodular function $f(S)$ to produce a relevant, well-covered, and non-redundant summary using the modified greedy algorithm for submodular function (Lin and Bilmes, 2010).

The reason behind choosing this algorithm is that a solution is guaranteed to be within a constant factor $(1 - 1/\sqrt{e})$ of the optimal solution when the objective function is monotone submodular. Since the scoring function $f(s)$ of our proposed summarizer is non-decreasing monotone submodular, we thus use the following greedy algorithm to obtain the near optimal solution.

³<http://jmlr.org/papers/volume5/lewis04a/a11-smart-stop-list/english.stop>

⁴<https://code.google.com/archive/p/word2vec/>

Algorithm 1 A Greedy algorithm for maximizing the objective function

Require: A minimization LP in standard form.

Ensure: Integral solution, IR1 to the LP.

```

1:  $S \leftarrow \emptyset, M \leftarrow \{1, \dots, N\}$ 
2: while  $M \neq \emptyset$  do
3:    $q \leftarrow \underset{p \in M}{\text{argmax}} \frac{f(S \cup \{p\}) - f(S)}{(c_p)^r}$ 
4:   if  $\sum_{j \in S} C_j + C_q \leq B_{max}$  and  $f(S \cup \{q\}) - f(S) \geq 0$  then
5:      $S \leftarrow S \cup \{q\}$ 
6:   end if
7:    $M \leftarrow M \setminus \{q\}$ 
8: end while
9:  $t^* \leftarrow \underset{t \in \{1, \dots, N\}, c_t \leq B_{max}}{\text{argmax}} f(\{t\})$ 
10: if  $f(t^*) > f(S)$  then
11:   return  $t^*$ 
12: else return  $S$ 
13: end if

```

4.2 Query-focused Summarization

For the query-focused summarization phase, we propose the following objective function:

$$f(S) = \alpha r(S) + \Upsilon q(S) + \Lambda h(S) \quad (6)$$

where, $r(S)$ measures summary’s importance quality, $q(S)$ measures summary’s query relevance quality, $h(S)$ measures summary’s non-redundancy quality and α , Υ , and Λ are non-negative trade-off coefficients which can be tuned empirically⁵.

We keep the importance and non-redundancy reward function similar to the generic summarizer described in the previous section. In addition, we design a query relevance objective function which considers the two important aspects: (1) how related summary sentences are with the query?, and (2) how much query dependent information is covered in the summary?

$$q(S) = \psi \cdot \sum_{j \in S} \text{Sim}(q, s_j) + \theta \cdot n_{j,q} \quad (7)$$

where, $\text{Sim}(i, j)$ is the similarity between summary sentence j and query q , here similarity means the cosine similarity of the average word vectors obtained from Word2Vec (Mikolov et al., 2013) for the query and the summary sentence. $n_{j,q}$ is the number of query terms present in the

⁵The values for the coefficients are 1.0, 10.0, and 5.0 for α , Υ , and Λ respectively, as found empirically using the development set DUC-2006 during the experiments.

summary sentence j . ψ , and θ are non-negative trade-off coefficients which have been tuned empirically during the experiments⁶.

5 Experiments

To evaluate our generic and query-focused summaries, we use DUC-2004 and DUC-2007 datasets, respectively. We perform some basic pre-processing on all the documents such as tokenization, part-of-speech tagging and document coreference resolution using Stanford CoreNLP (Manning et al., 2014). We also use Porter’s stemmer (Porter, 1999) for stemming all the words and remove all the stop words from the smart stop words list⁷. For query-focused summarization, we use word vectors from Word2Vec (Mikolov et al., 2013) which allows us to obtain better similarity scores between the sentences and the queries. We evaluate our system generated summaries using the automatic evaluation toolkit ROUGE version 1.5.5 (Lin, 2004).

We compare the results of our systems (i.e., document shrinking + summarization or document summarization + shrinking) with other state-of-the-art generic summarization methods. The comparison is shown in Table 1 where we report the values of ROUGE-1 recall and F-1 measure⁸ of different approaches. From the table, we can see that our generic multi-document summarizer (document shrinking + summarization) significantly outperforms those systems in all measures. This result suggests the effectiveness of sentence compression and merging phase in our system. It also shows the effectiveness of using semantic similarity measures to select important sentences in the summary. Moreover, our system also uses a separate redundancy function which also helps to generate summaries with less redundancy compared to the systems which only concentrate on summary’s coverage and relevance. These results also confirm that the proposed strategy can improve summary quality.

We compare our query-focused summarizer with other state-of-the-art query summarization methods. Table 2 shows the comparison in terms

⁶The values for the query relevance coefficients are 4.0 and 2.0 for ψ and θ respectively, as found empirically using the development set DUC-2006 during the experiments.

⁷<http://jmlr.org/papers/volume5/lewis04a/a11-smart-stop-list/english.stop>

⁸ROUGE runtime arguments for DUC-2004: ROUGE -a -c 95 -b 665 -m -n 4 -w 1.2

Systems	R-1	F-1
Best system in DUC-04 (peer 65)	0.3828	0.3794
(Takamura and Okumura, 2009)	0.385	-
(Lin and Bilmes, 2011)	0.3935	0.389
(McDonald, 2007)	0.362	0.338
(Wang et al., 2009)	0.3907	-
Document Shrinking + Summarization	0.4127	0.4133
Document Summarization + Shrinking	0.3874	0.3882

Table 1: Results on DUC-2004 Datasets

Systems	R-2	F-2
Best system in DUC-07 (peer 15)	0.1245	0.1229
(Lin and Bilmes, 2011)	0.1238	0.1233
(Toutanova et al., 2007)	0.1189	0.1189
(Haghighi and Vanderwende, 2009)	0.118	-
Document Shrinking + Summarization	0.1258	0.1264
Document Summarization + Shrinking	0.1133	0.1149

Table 2: Results on DUC-2007 Datasets

of ROUGE scores⁹ between our system and the best performing systems. From the table, we can say that our query-focused multi-document summarizer (document shrinking + summarization) outperforms the best-known systems in DUC-2007. It is notable that the best system in DUC-2007 takes the topic title as a query and uses Yahoo search engine to get a ranked set of retrieved documents which were used later to calculate the query relevance score (Pingali et al., 2007). However, our system is totally unsupervised and does not use any external source for the summary generation.

6 Conclusion

In this paper, we proposed a new summarization framework using different submodular functions with deep semantic features and abstraction-based methods. Abstraction-based methods help the system to obtain concise and more informative candidate summary sentences. We selected the best sentences for the summary by maximizing the submodular objective function. The empirical results show that our generic and query-focused summarization model outperform the state-of-the-art systems.

Acknowledgments

This research was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the University of Lethbridge.

⁹ROUGE runtime arguments for DUC-2007: ROUGE -n 2 -x -m -2 4 -u -c 95 -r 1000 -f A-p 0.5-t 0-d

References

- R. Barzilay and K. R. McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.
- T. Berg-Kirkpatrick, D. Gillick, and D. Klein. 2011. Jointly learning to extract and compress. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 481–490. Association for Computational Linguistics.
- L. Bing, P. Li, Y. Liao, W. Lam, W. Guo, and R. J. Passonneau. 2015. Abstractive multi-document summarization via phrase selection and merging. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*.
- F. Boudin, H. Mougard, and B. Favre. 2015. Concept-based summarization using integer linear programming: From concept pruning to multiple optimal solutions. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1918. Association for Computational Linguistics.
- J. Carbonell and J. Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM.
- Y. Chali and M. Uddin. 2016. Multi-document summarization based on atomic semantic events and their temporal relationships. In *Advances in Information Retrieval*, pages 366–377. Springer.
- J. Cheng and M. Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany. Association for Computational Linguistics.
- J. C. Cheung and G. Penn. 2014. Unsupervised sentence enhancement for automatic summarization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 775–786. Association for Computational Linguistics.
- S. Chopra, M. Auli, A. M. Rush, and S. Harvard. 2016. Abstractive sentence summarization with attentive recurrent neural networks. *Proceedings of NAACL-HLT16*, pages 93–98.
- G. Cornuejols, M. L. Fischer, and G. L. Nemhauser. 1977. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Science*, 23(8):789–810.
- A. Dasgupta, R. Kumar, and S. Ravi. 2013. Summarization through submodularity and dispersion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics ACL*, pages 1014–1022.
- G. Erkan and D. R. Radev. 2004. Lexrank: graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, pages 457–479.
- C. Fellbaum, editor. 1998. *Wordnet: An Electronic Database*. MIT Press.
- K. Filippova, E. Alfonseca, C. A. Colmenares, L. Kaiser, and O. Vinyals. 2015. Sentence compression by deletion with lstms. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 360–368.
- K. Ganesan, C. Zhai, and J. Han. 2010. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd international conference on computational linguistics*, pages 340–348. Association for Computational Linguistics.
- A. Haghighi and L. Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370. Association for Computational Linguistics.
- K. Hong and A. Nenkova. 2014. Improving the estimation of word importance for news multi-document summarization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 712–721. Association for Computational Linguistics.
- K. Knight and D. Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *Proceedings of the 17th National Conference on Artificial Intelligence*, Austin.
- H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, and D. Jurafsky. 2013. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916.
- C.-Y. Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, Workshop on Text Summarization Branches Out*.
- H. Lin and J. Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Proceedings of the 2010 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 912–920. Association for Computational Linguistics.

- H. Lin and J. Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics.
- C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics System Demonstrations*.
- A. Martins and N. A. Smith. 2009. Summarization with a joint model for sentence extraction and compression. In *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*, pages 1–9. Association for Computational Linguistics.
- R. McDonald. 2007. A study of global inference algorithms in multi-document summarization. In *European Conference on Information Retrieval*, pages 557–564. Springer.
- R. Mihalcea and P. Tarau. 2004. Textrank: Bringing order into texts. In *Proceedings of EMNLP 2004*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.
- T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality. In *In Advances in neural information processing systems*, pages 3111–3119.
- H. Morita, T. Sakai, and M. Okumura. 2011. Query snowball: a co-occurrence-based approach to multi-document summarization for question answering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 223–229. Association for Computational Linguistics.
- R. Nallapati, F. Zhai, and B. Zhou. 2017. Summarunner: An interpretable recurrent neural network model for extractive summarization. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, San Francisco. AAAAI Press.
- D. Parveen, H. M. Ramsel, and M. Strube. 2015. Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1949–1954. Association for Computational Linguistics.
- D. Parveen and M. Strube. 2015. Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, page 12981304.
- P. Pingali, K. Rahul, and V. Varma. 2007. Iiit hyderabad at duc 2007. In *Proceedings of the Document Understanding Conference*.
- M.F. Porter. 1999. An algorithm for suffix stripping. *Program*, 14(3):130–137.
- M. A. Rush, S. Chopra, and J. Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389. Association for Computational Linguistics.
- H. Takamura and M. Okumura. 2009. Text summarization model based on maximum coverage problem and its variant. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 781–789. Association for Computational Linguistics.
- K. Thadani and K. McKeown. 2013. Supervised sentence fusion with single-stage inference. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 1410–1418.
- K. Toutanova, C. Brockett, M. Gamon, J. Jagarlamudi, H. Suzuki, and L. Vanderwende. 2007. The pythy summarization system: Microsoft research at duc 2007. In *Proceedings of the Document Understanding Conference*.
- D. Wang, S. Zhu, T. Li, and Y. Gong. 2009. Multi-document summarization using sentence-based topic models. In *Proceedings of the ACL-IJCNLP 2009 Conference*, pages 297–300. Association for Computational Linguistics.
- L. Wang and W. Ling. 2016. **Neural network-based abstract generation for opinions and arguments**. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57, San Diego, California. Association for Computational Linguistics.