A Redundancy-Aware Sentence Regression Framework for Extractive Summarization

Pengjie Ren[1*] **Furu Wei**[2] **Zhumin Chen**[1†] **Jun Ma**[1] **Ming Zhou**[2]
jay.ren@outlook.com, {chenzhumin,majun}@sdu.edu.cn, {fuwei,mingzhou}@microsoft.com
[1]Department of Computer Science and Technology, Shandong University / Jinan, China
[2]Microsoft Research Asia / Beijing, China

Abstract

Existing sentence regression methods for extractive summarization usually model sentence importance and redundancy in two separate processes. They first evaluate the importance $f(s)$ of each sentence s and then select sentences to generate a summary based on both the importance scores and redundancy among sentences. In this paper, we propose to model importance and redundancy simultaneously by directly evaluating the relative importance $f(s|S)$ of a sentence s given a set of selected sentences S. Specifically, we present a new framework to conduct regression with respect to the relative gain of s given S calculated by the ROUGE metric. Besides the single sentence features, additional features derived from the sentence relations are incorporated. Experiments on the DUC 2001, 2002 and 2004 multi-document summarization datasets show that the proposed method outperforms state-of-the-art extractive summarization approaches.

1 Introduction

Sentence regression is one of the branches of extractive summarization methods that achieves state-of-the-art performances (Cao et al., 2015b; Wan et al., 2015) and is commonly used in practical systems (Hu and Wan, 2013; Wan and Zhang, 2014; Hong and Nenkova, 2014). Existing sentence regression methods usually model sentence importance and sentence redundancy in two separate processes, namely sentence ranking and sentence selection. Specifically, in the sentence ranking process, they evaluate the importance $f(s)$ of each sentence s with a ranking model (Osborne, 2002; Conroy et al., 2004; Galley, 2006; Li et al., 2007) through either directly measuring the salience of sentences (Li et al., 2007; Ouyang et al., 2007) or firstly ranking words (or bi-grams) and then combining these scores to rank sentences (Lin and Hovy, 2000; Yih et al., 2007; Gillick and Favre, 2009; Li et al., 2013). Then, in the sentence selection process, they discard the redundant sentences that are similar to the already selected sentences.

In this paper, we propose a novel regression framework to directly model the relative importance $f(s|S)$ of a sentence s given the sentences S. Specifically, we evaluate the relative importance $f(s|S)$ with a regression model where additional features involving the sentence relations are incorporated. Then we generate the summary by greedily selecting the next sentence which maximizes $f(s|S)$ with respect to the current selected sentences S. Our method improves the existing regression framework from three aspects. First, our method is redundancy-aware by considering importance and redundancy simultaneously instead of two separate processes. Second, we treat the scores computed using the official evaluation tool as the groundtruth and find that our method has a higher upper bound. Third, there is no manually tuned parameters, which is more convenient in practice. We carry out experiments on three benchmark datasets from DUC 2001, 2002, and 2004 multi-document summarization tasks. Experimental results show that our method achieves the best performance in terms of ROUGE-2 recall metric and outperforms state-of-the-art extractive summarization approaches on all three datasets.

[*] This work was done during the internship of the first author at Microsoft Research Asia.
[†] Corresponding author.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 33–43, Osaka, Japan, December 11-17 2016.

2 Framework

2.1 Background

Formally, given a sentence set (from one or multiple documents) $D \in C$, extractive summarization tries to select a sentence set S^* as the summary that maximizes an utility function $f(S)$ with respective to the length limit l, Existing sentence regression methods usually model the importance of each sentence independently (Osborne, 2002; Galley, 2006; Li et al., 2007). Then, they use a threshold parameter to control the redundancy (Cao et al., 2015b; Galanis et al., 2012) when selecting sentences with the Greedy algorithm or Integer Linear Programming (ILP) algorithm (Cao et al., 2015a). The framework for these regression methods can be formulated as follows.

$$f(s|S) = \begin{cases} f(s) & 1 - SIM(s, S) \geq t \\ 0 & 1 - SIM(s, S) < t \end{cases} \tag{1}$$

where S is the set of already selected sentences, $f(s)$ models the importance of sentence s. $SIM(s, S)$ evaluates the similarity of sentence s with the current generated summary S. Usually, $SIM(s, S) = \frac{\textit{bi-gram-overlap}(s,S)}{Len(s)}$, which is the bi-gram overlap ratio. $Len(s)$ is the length of s. t is a threshold parameter used to control the redundancy, which is usually set heuristically.

2.2 Our Framework

In this paper, we propose to directly model the relative importance $f(s|S)$ instead of the independent importance of each sentence $f(s)$. Specially, we model the importance of s given the sentences S as follows:

$$f(s|S) = \min_{s' \in S} f(s|s') \tag{2}$$

which considers the minimum relative importance of sentence s with respect to each sentence of S. $f(s|s')$ models the relative importance of sentence s given sentence s', which makes Equation 2 a redundancy-aware framework.

When generating summaries, we select the first sentence by treating $s' = \emptyset$ or using a $f(s)$ model. Then, we select the rest summary sentences with the following greedy algorithm:

$$s^* = \arg \max_{s \subset D \setminus S} \min_{s' \in S} f(s|s') \tag{3}$$

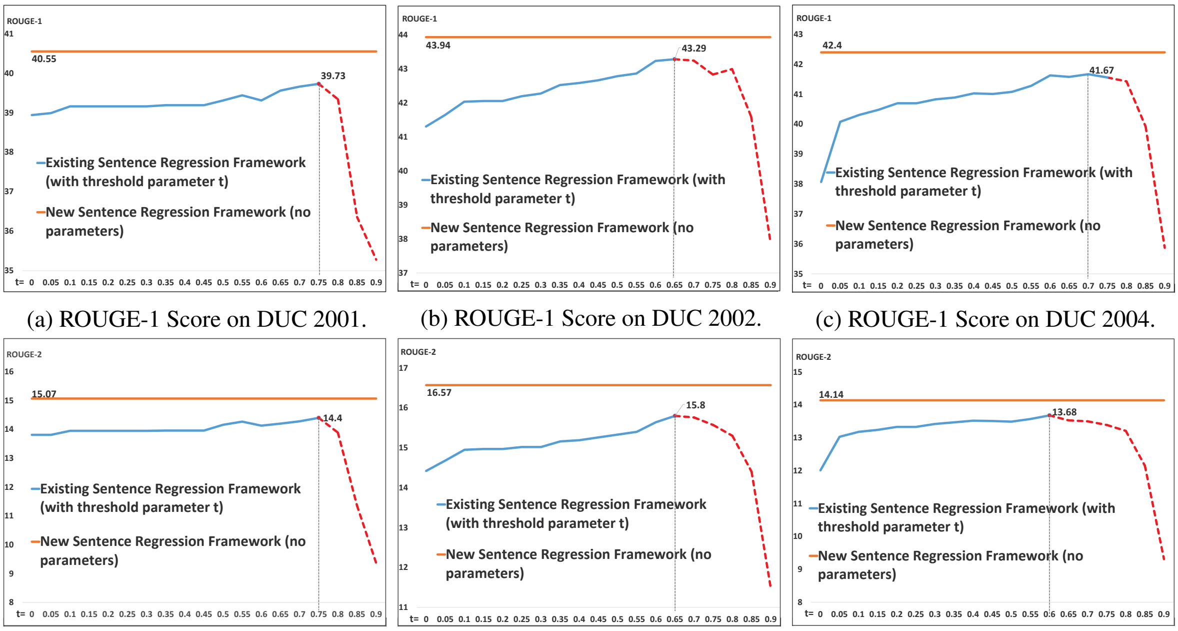
The algorithm starts with the first selected sentence. In each step, a new sentence is added to the summary that results in the maximum relative increase according to Equation 3. The algorithm terminates when the summary length constraint is reached.

Next we conduct experiments to analyze the upper bounds of the new framework compared with the existing framework (Equation 1). To this end, we compute $f(s)$ and $f(s|s')$ as follows:

$$\begin{aligned} & f(s) = \textit{ROUGE-}2(s|S_{ref}) \\ & f(s|s') = f(\{s, s'\}) - f(s') = \textit{ROUGE-}2(\{s, s'\}|S_{ref}) - \textit{ROUGE-}2(s'|S_{ref}) \end{aligned} \tag{4}$$

where S_{ref} is one or several summaries written by people. The ROUGE-2 recall metric gives a score to a set of sentences with respective to the human written summaries. We compute $f(s|s')$ as the total gain of s and s' ($f(\{s, s'\})$) subtracted by the individual gain of s' ($f(s')$). Equation 4 can be seen as the groundtruth computation of $f(s)$ and $f(s|s')$.

The experimental upper bounds of different frameworks are shown in Figure 1. Similar results of ROUGE-1 and ROUGE-2 are achieved on all three benchmark datasets from DUC 2001, 2002 and 2004. The *advantages* of the new framework (Equation 2) are three-fold compared with the framework of Equation 1. First, there is no parameter to be tuned manually. By comparison, Equation 1 has a threshold parameter t, which is very sensitive around the best performance, as shown in the red dashed line parts of Figure 1. Second, the new framework has a higher upper bound, which means there is a bigger potential



(a) ROUGE-1 Score on DUC 2001.

(b) ROUGE-1 Score on DUC 2002.

(c) ROUGE-1 Score on DUC 2004.

(d) ROUGE-2 Score on DUC 2001.

(e) ROUGE-2 Score on DUC 2002.

(f) ROUGE-2 Score on DUC 2004.

Figure 1: Experimental Upper bounds of our sentence regression framework and existing sentence regression framework.

for improvement. Finally, besides single sentence features, additional features involving the relations of two sentences can be extracted to improve the regression performance.

The new proposed framework also has some *challenges*. First, the groundtruth of $f(s|s')$ is usually unavailable for many tasks. Fortunately, in the text summarization task, the groundtruth of $f(s|s')$ can be computed according to Equation 4. Second, the number of training instances is $O(|C||D|^2)$ ($O(|C||D|)$ for Equation 1). We come up with two ways to speed up the training process in the next session.

3 Implementation

3.1 Objective Function

We implement $f(s|s')$ with MultiLayer Perceptron (MLP) (Ruck et al., 1990; Gardner and Dorling, 1998).

$$f(s|s') = MLP\left(\Phi(s|s')|\theta\right) \tag{5}$$

where $\Phi(s|s')$ is the set of features and θ is the parameters to be learned.

We use the standard Mean Square Error (MSE) as the loss function as follows:

$$\begin{aligned} L(\theta) &= \frac{1}{|C||D|(|D|-1)} \sum_{D \in C} \sum_{s \in D} \sum_{\substack{s' \in D;\\ s' \neq s}} Err(s|s') \\ Err(s|s') &= \left(MLP\left(\Phi(s|s')|\theta\right) - ROUGE(s|s', S_{ref})\right)^2 \\ ROUGE(s|s', S_{ref}) &= ROUGE\text{-}2(\{s, s'\}|S_{ref}) - ROUGE\text{-}2(s'|S_{ref}) \end{aligned} \tag{6}$$

We use ROUGE-2 recall as the groundtruth score due to its high capability of evaluating automatic summarization systems (Owczarzak et al., 2012).

The s' in $f(s|s')$ should mainly refer to the sentences that have a big potential to be selected into the summary. To this end, we do not have to treat each sentence in D as s' during training. We can accelerate the training process by generating a set of sentences S' from D. We come up with two ways as shown in Algorithm 1. The first way is using the greedy strategy (Line 4 of Algorithm 1). Specifically, for each training episode of sentence s, we use the current model to generate the summary with greedy algorithm as a part of the S'. We refer to this part as S'_1. The advantage is that S'_1 is adaptively generated with respective to the training status of the model. The second way is randomly sampling a small set of s' with respect to its groundtruth ROUGE score (Line 6 of Algorithm 1). Specifically, for each training episode of sentence s, we sample a small set S'_2 according to the following rule:

$$\begin{cases} NotSelected & rnd(0,1) > 0.05 * ROUGE\text{-}2(s) + 0.05 \\ Selected & Otherwise \end{cases} \tag{7}$$

where $rnd(0,1)$ generates random number from a uniform distribution within the range $[0,1]$. $ROUGE\text{-}2(s)$ is normalized to $[0,1]$. Each sentence is selected with at least 5% probability and sentences with higher ROUGE scores have higher probabilities. Different probabilities will influence the speed of the training process but will not make much difference in the final results according to our experiments. We use the randomly sampled S'_2 to avoid the premature convergence caused by S'_1. Finally, $S' = S'_1 \bigcup S'_2$. In this way, the number of training instances is $O(|C||D||S'|)$ while originally it is $O(|C||D|^2)$, where C is the set of all D in the training corpus. Note that $|S'|$ is a very small number compared to $|D|$.

Algorithm 1 The adaptive & randomized training.

```
Input:
    Training corpus, $C$;
    Max iterations, $N$;
Output:
    Model parameters, $\theta$;
 1: Randomly initialize the parameters $\theta$;
 2: for $i = 1; i < N; i{+}{+}$ do
 3:     for each $D$ such that $D \in C$ do
 4:         Generate $S'_1$ greedily according to Equation 3;
 5:         for each sentence $s$ such that $s \in D$ do
 6:             Generate $S'_2$ randomly according to Equation 7;
 7:             for each $s'$ such that $s' \in S'_1 \bigcup S'_2, s' \neq s$ do
 8:                 Make forward and backward propagation w.r.t the loss $L(\theta)$ (Equation 6);
 9:                 Update the model parameters $\theta$;
10:             end for
11:         end for
12:     end for
13:     if $\theta$ converges then
14:         break;
15:     end if
16: end for
17: return $\theta$;
```

3.2 Feature

We employ two groups of features in terms of sentence importance and redundancy, namely Sentence Importance Features and Sentence Relation Features. The former are widely studied by existing methods (Gupta et al., 2007; Li et al., 2007; Aker et al., 2010; Ouyang et al., 2011; Galanis et al., 2012; Hong et al., 2015). However, to our knowledge, the latter are firstly incorporated into a regression model in this paper. Details of used features are listed in Table 1. We use Sentence Importance Features to model the independent sentence importance of s. $Len(s)$, $Position(s)$, $Stop(s)$, $TF(s)$ and $DF(s)$ are commonly used features. Embedding feature $Emb(s)$ is an effective feature that encodes the sentence content which can be seen as summary prior nature of the sentence (Cao et al., 2015b). We use Sentence Relation Features to evaluate the content overlap between s and s'. $Match\text{-}P(s \cap s')$ and $Match\text{-}R(s \cap s')$ evaluate the ratio of the overlap words, while $TF(s \cap s')$, $DF(s \cap s')$ and $Stop(s \cap s')$ evaluate the importance of the overlap words. $Cos(s, s')$ evaluates the exact match similarity while $Emb\text{-}Cos(s, s')$ evaluates the meaning match similarity. All features in Table 1 are basic features commonly used in summarization.

Features	Formulations	Descriptions
Sentence Importance Features	$Len(s)$	Length of s
	$Position(s)$	Position of s in its document
	$Stop(s) = \frac{stop\text{-}count(s)}{Len(s)}$	Stop words ratio of s
	$TF(s) = \frac{\sum_{w \in s} GTF(w)}{Len(s)}$	Average Term Frequency $GTF(w)$ is the Global Term Frequency
	$DF(s) = \frac{\sum_{w \in s} DF(w)}{Len(s)}$	Average Document Frequency
	$Emb(s) \frac{\sum_{w \in s} Emb(w)}{Len(s)}$	Average Word Embedding $Emb(w)$ is the word embedding
Sentence Relation Features	$Match\text{-}P(s, s') = \frac{Match(s,s')}{Len(s)}$	Term match precision $Match\text{-}P(s, s') = 0$ if $s \cap s' = \emptyset$
	$Match\text{-}R(s, s') = \frac{Match(s,s')}{Len(s')}$	Term match recall $Match\text{-}R(s, s') = 0$ if $s \cap s' = \emptyset$
	$TF(s, s') = \frac{Len(s \cap s')}{\sum_{w \in s \cap s'} GTF(w)}$	Average Global Term Frequency of overlap $s \cap s'$ $TF(s, s') = 0$ if $s \cap s' = \emptyset$
	$DF(s, s') = \frac{Len(s \cap s')}{\sum_{w \in s \cap s'} DF(w)}$	Average Document Frequency of overlap $s \cap s'$ $DF(s, s') = 0$ if $s \cap s' = \emptyset$
	$Stop(s, s') = 1 - \frac{Stop\text{-}Count(s \cap s')}{Len(s \cap s')}$	Stop words ratio of overlap $s \cap s'$ $Stop(s, s') = 0$ if $s \cap s' = \emptyset$
	$Cos(s, s')$ $= Cosine(GTF(s), GTF(s'))$	Cosine of Global Term Frequency vector
	$Emb\text{-}Cos(s, s')$ $= Cosine(Emb(s), Emb(s'))$	Cosine of average embedding vector

Table 1: Summary of features

4 Experiment

4.1 Experimental Setup

Datasets. The benchmark evaluation corpora for summarization are the ones published by the Document Understanding Conferences (DUC[1]). We focus on the generic multi-document summarization task, so we carried out all our experiments on DUC 2001, 2002 and 2004 datasets. The documents are all from the news domain and are grouped into various thematic clusters. For each document set, we concatenated all the articles and split them into sentences using the tool provided with the DUC 2003 dataset. We train the model on two years' data and test it on the other year.
Evaluation Metric. ROUGE metrics are the official metrics of the DUC extractive summarization tasks (Rankel et al., 2013). We use the official ROUGE tool[2] to evaluate the performance of the baselines as well as our approach (Lin, 2004). The parameter of length constraint is "-l 100" for DUC 2001/2002, and "-b 665" for DUC 2004. We take ROUGE-2 recall as the main metric for comparison because Owczarzak et al. prove its high capability of evaluating automatic summarization systems (Owczarzak et al., 2012).
Comparison Methods. The comparison methods used in the experiments are listed as follows.

- LexRank: State-of-the-art summarization model (Erkan and Radev, 2004).
- ClusterHITS: State-of-the-art results on DUC 2001 (Wan and Yang, 2008).
- ClusterCMRW: State-of-the-art results on DUC 2002 (Wan and Yang, 2008).
- REGSUM[3]: State-of-the-art results on DUC 2004 (Hong and Nenkova, 2014).
- R2N2_GA/R2N2_ILP: State-of-the-art results on DUC 2001/2002/2004 (Cao et al., 2015a) with a neural network regression model.
- PriorSum: To our knowledge, the best results on DUC 2001, 2002 and 2004 using regression approaches (Cao et al., 2015b).
- SR (Sentence Regression): Evaluate sentence importance with MLP and the Sentence Importance Features in Table 1 and select the top ranks as the summary (without handling redundancy).

[1] http://duc.nist.gov/

[2] ROUGE-1.5.5 with options: -n 2 -m -u -c 95 -x -r 1000 -f A -p 0.5 -t 0

[3] REGSUM truncates a summary to 100 words.

	DUC 2001		DUC 2002		DUC 2004	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
BestSentence	37.32	10.44	39.75	11.60	40.36	11.68
Strategy 1	36.31	8.49	37.80	9.61	39.60	10.57
Strategy 2	36.32	8.52	37.82	9.26	38.75	10.19

Table 2: First sentence selection strategies

- t-SR (threshold t based Sentence Regression): Evaluate sentence importance with MLP and the Sentence Importance Features in Table 1 and generate the summary with greedy by directly discarding the redundant sentence according to Equation 1.
- RASR (**R**edundancy-**A**ware **S**entence **R**egression): The proposed method in this paper.

Note that for the methods with the parameter t, we tried all values of ranging from 0 to 1 with a step size of 0.05. The final value of t on each dataset is decided by 3-fold cross validation on the training datasets.

Model Configuration. The word embedding used in this paper is trained on the English Wikipedia Corpus[4] with Google's Word2Vec tool[5]. The dimension is 300. We use 4 hidden layers MLP with tanh activation function and the sizes of the layers are $[300, 200, 100, 1]$. To update the weights of MLP, we apply the diagonal variant of AdaGrad with mini-batches. We set the mini-batch size to 20.

4.2 Results and Analysis

First Sentence Selection. Remember that when generating a summary, our method first selects the first sentence then greedily selects the rest sentences with respective to $f(s|S)$. We tried two strategies to select the first sentence with RASR. Strategy 1: treating RASR as an united model by setting the Sentence Relation Features to zero when fitting $f(s)$ during training period or selecting the first sentence during test period. Strategy 2: treating RASR as two models that fit $f(s)$ and $f(s|S)$ respectively. The former is used to select the first sentence and the latter is used to select the rest sentences. We also use the sentence that gets the highest ROUGE-2 score as the first sentence as a comparison, namely BestSentence. The results are shown in Table 2. As expected, BestSentence is much better than Strategy 1 and Strategy 2, which means selecting a better first sentence will greatly improve the performance of RASR. It does not make too much difference whether using Strategy 1 or Strategy 2. We report the results of Strategy 1 to compare with the baseline methods in Table 3.

Performance Analysis. As shown in Table 3, the bold face indicates the best performance. Generally, our method RASR achieves the best performance in terms of ROUGE-2 metric on all three datasets. The improvement of ROUGE-2 on DUC 2001 is significant with p-value < 0.05 compared with LexRank, SR and t-SR. Although ClusterHITS and ClusterCMRW get higher ROUGE-1 scores, their ROUGE-2 scores are much lower. In contrast, our method works quite stably.

The improvements of our method come from two aspects. First, it is effective to model sentence importance and redundancy simultaneously with multiple nonlinear function transformations. This can be reflected by the following comparison experiments. SR does not handle redundancy at all, so it achieves bad performance especially on the DUC 2004 corpus. The other methods in Table 3 model sentence importance and redundancy in two separate processes by first ranking the sentences and then discarding the redundant ones whose bi-gram overlap ratio is larger than a threshold parameter. Although we tune the threshold parameter carefully, RASR still outperforms them. Second, effective features involving the sentence relations (i.e., Sentence Relation Features) are considered which cannot be incorporated by the baseline methods.

[4] https://en.wikipedia.org/wiki/Wikipedia:Database_download

[5] https://code.google.com/archive/p/word2vec/

	System	ROUGE-1	ROUGE-2
DUC 2001	Peer T	33.03	7.86
	ClusterHITS*	**37.42**	6.81
	LexRank	33.43	6.09
	R2N2_GA*	35.88	7.64
	R2N2_ILP*	36.91	7.87
	PriorSum*	35.98	7.89
	SR	35.34	7.67
	t-SR	35.41	7.76
	RASR	36.31	**8.49**
DUC 2002	Peer 26	35.15	7.64
	ClusterCMRW*	**38.55**	8.65
	LexRank	35.29	7.54
	R2N2_GA*	36.84	8.52
	R2N2_ILP*	37.96	8.88
	PriorSum*	36.63	8.97
	SR	36.70	8.59
	t-SR	37.49	8.95
	RASR	37.80	**9.61**
DUC 2004	Peer 65	37.88	9.18
	REGSUM*	38.57	9.75
	LexRank	37.87	8.88
	R2N2_GA*	38.16	9.52
	R2N2_ILP*	38.78	9.86
	PriorSum*	38.91	10.07
	SR	35.76	8.73
	t-SR	38.36	9.98
	RASR	**39.60**	**10.57**

Peer T/Peer 26/Peer 65 are the original results on DUC 2001/2002/2004 respectively. We cite the scores of some systems from their papers, indicated with the sign "*".

Table 3: Comparison results (%) on DUC datasets

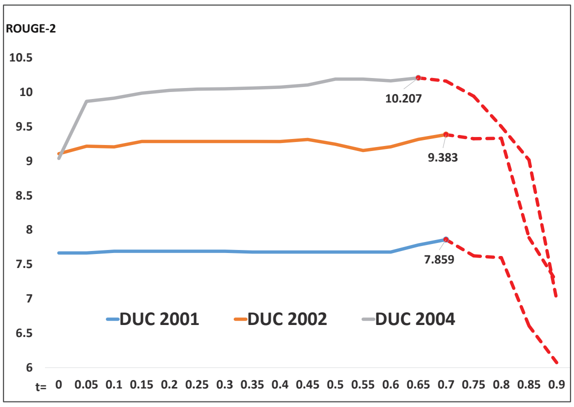
Parameter Sensitiveness. We present the ROUGE-2 performance of t-SR with the threshold parameter t ranging from 0 to 0.9 with a step size of 0.05 shown in Figure 1 and 2a. The best achieved performances of the groundtruth implementation are around 0.75, 0.65, 0.6 (Figure 1) while the best achieved performances in practice are around 0.7, 0.7, 0.65 (Figure 2a). t is still very sensitive around the best performance, as shown in the red dashed line in both Figure 1 and 2a.

Training Convergence. In order to speed up the training process of RASR, we randomly sample some pairwise training instances with Equation 7 for training of RASR. We want to know whether this will influence the convergence of RASR, so we present the decrease of loss with respect to training iterations in Figure 2b. We find that the random sampling has little influence on the convergence of RASR with t-SR as a comparison.

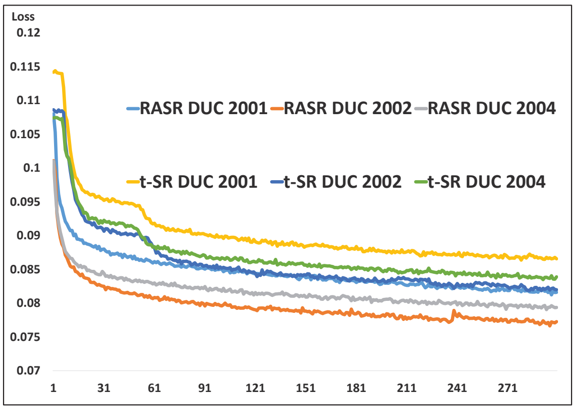
5 Related Work

Existing work on extractive summarization can be divided into two categories: unsupervised and supervised.

Two most famous unsupervised frameworks are Centroid based and Maximum Marginal Relevance based. Centroid-based methods evaluate the sentence centrality as its importance (Mihalcea, 2004). Radev et al. first propose to model cluster centroids in their summarization system, MEAD (Radev et al., 2000; Radev et al., 2004). Then LexRank (or TextRank) is proposed to compute sentence importance



(a) Sensitiveness of the parameter t of t-SR.



(b) Loss decrease with respect to training iterations.

based on the concept of eigenvector centrality in a graph of sentence similarities (Erkan and Radev, 2004; Mihalcea and Tarau, 2004). Due to its expansibility and flexibility, centroid-based methods have a lot of extensions. Wan et al. propose several centroid-based approaches for different summarization tasks, e.g., cross-language summarization, etc (Wan, 2008; Wan and Xiao, 2009; Wan, 2011). Maximum Marginal Relevance (MMR) based methods consider the linear trade-off between relevance and redundancy (Carbonell and Goldstein, 1998). Goldstein et al. first extend MMR to support extractive summarization by incorporating additional information (Goldstein et al., 2000). McDonald achieves good results by reformulating this as a knapsack packing problem and solving it using ILP (McDonald, 2007). Later Lin and Bilmes propose a variant of MMR framework which maximizes an objective function that considers the linear trade-off between coverage and redundancy terms (Lin and Bilmes, 2010; Lin and Bilmes, 2011).

Supervised methods model the extractive summarization task from various perspectives. Kupiec et al. train a naive-Bayes classifier to decide whether to include a particular sentence in the summary or not. (Kupiec et al., 1995). Li et al. evaluate the sentence importance with support vector regression, then a simple rule-based method is applied for removing redundant phrases (Li et al., 2007). Gillick and Favre evaluate bi-grams importance and then use these scores to evaluate sentence importance and redundancy with a linear combination (Gillick and Favre, 2009). Sipos et al. propose a structural SVM learning approach to learn the weights of feature combination using the MMR-like submodularity function proposed by Lin and Bilmes (Lin and Bilmes, 2010). Cao et al. evaluate the sentence importance with a neural regression model, then they remove the redundant sentence larger than a threshold parameter during greedy algorithm (Cao et al., 2015b). In another paper, they remove the redundant sentence by adding a redundancy constraint to the ILP objective which restricts the bi-gram redundancy of the selected sentences smaller than a threshold (Cao et al., 2015a).

In all above extractive summarization methods, redundancy is mainly considered in two ways. The first way is measuring the importance of each sentence then explicitly removing the redundant sentence larger than a threshold parameter during the sentence selection process. Another way is linearly substracting the sentence redundancy score or scoring the redundant parts with low weights. To the best of our knowledge, none of them studies the summarization task and models redundancy from the perspective of this paper.

6 Conclusion and Future Work

This paper presents a novel sentence regression framework to conduct regression with respect to the relative importance $f(s|S)$ of sentence s given a set of sentences S. Additional features involving the sentence relations are incorporated. We conduct experiments on three DUC benchmark datasets. Generally, our approach achieves the best performance in terms of ROUGE metrics compared with state-of-the-art approaches.

We believe our work can be advanced and extended from many different perspectives. First, more features can be designed especially those involving the relations of two sentences. Second, the results can be further improved by exploring better strategies to select the first sentence. Third, the framework can be extended to other tasks, e.g., query-focused summarization, which can be achieved by introducing query-related features.

Acknowledgement

This work is supported by the Natural Science Foundation of China (61672322, 61672324, 61272240), Microsoft Fund (FY14-RES-THEME-025), the Natural Science Foundation of Shandong Province (ZR2012FM037), the Excellent Middle-Aged and Youth Scientists of Shandong Province (B-S2012DX017) and the Fundamental Research Funds of Shandong University.

References

Ahmet Aker, Trevor Cohn, and Robert Gaizauskas. 2010. Multi-document summarization using a* search and discriminative training. In *Proceedings of EMNLP*, pages 482–491. Association for Computational Linguistics.

Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. 2015a. Ranking with recursive neural networks and its application to multi-document summarization. In *Proceedings of AAAI*, pages 2153–2159. AAAI Press.

Ziqiang Cao, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou, and Houfeng Wang. 2015b. Learning summary prior representation for extractive summarization. *Proceedings of ACL*, pages 829–833.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of SIGIR*, pages 335–336. ACM.

John M Conroy, Judith D Schlesinger, Jade Goldstein, and Dianne P Oleary. 2004. Left-brain/right-brain multi-document summarization. In *Proceedings of the DUC*.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, pages 457–479.

Dimitrios Galanis, Gerasimos Lampouras, and Ion Androutsopoulos. 2012. Extractive multi-document summarization with integer linear programming and support vector regression. In *Proceedings of COLING*, pages 911–926. Citeseer.

Michel Galley. 2006. A skip-chain conditional random field for ranking meeting utterances by importance. In *Proceedings of EMNLP*, pages 364–372. Association for Computational Linguistics.

Matt W Gardner and SR Dorling. 1998. Artificial neural networks (the multilayer perceptron)a review of applications in the atmospheric sciences. *Atmospheric environment*, 32(14):2627–2636.

Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, pages 10–18. Association for Computational Linguistics.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization*, pages 40–48. Association for Computational Linguistics.

Surabhi Gupta, Ani Nenkova, and Dan Jurafsky. 2007. Measuring importance and query relevance in topic-focused multi-document summarization. In *Proceedings of ACL*, pages 193–196. Association for Computational Linguistics.

Kai Hong and Ani Nenkova. 2014. Improving the estimation of word importance for news multi-document summarization. In *Proceedings of EACL*, pages 712–721.

Kai Hong, Mitchell Marcus, and Ani Nenkova. 2015. System combination for multi-document summarization. In *Proceedings of EMNLP*, pages 107–117. The Association for Computational Linguistics.

Yue Hu and Xiaojun Wan. 2013. Ppsgen: Learning to generate presentation slides for academic papers. In *Proceedings of IJCAI*. AAAI Press.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of SIGIR*, pages 68–73. ACM.

Sujian Li, You Ouyang, Wei Wang, and Bin Sun. 2007. Multi-document summarization using support vector regression. In *Proceedings of DUC*. Citeseer.

Chen Li, Xian Qian, and Yang Liu. 2013. Using supervised bigram-based ilp for extractive summarization. In *Proceedings of ACL*, pages 1004–1013. Citeseer.

Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 912–920. Association for Computational Linguistics.

Hui Lin and Jeff 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*, pages 510–520. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *Proceedings of COLING*, pages 495–501. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8.

Ryan McDonald. 2007. A study of global inference algorithms in multi-document summarization. *Advances in Information Retrieval*, pages 557–564.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. In *Proceedings of EMNLP*, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics.

Rada Mihalcea. 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of ACL*, page 20. Association for Computational Linguistics.

Miles Osborne. 2002. Using maximum entropy for sentence extraction. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 1–8. Association for Computational Linguistics.

You Ouyang, Sujian Li, and Wenjie Li. 2007. Developing learning strategies for topic-based summarization. In *Proceedings of CIKM*, pages 79–86. ACM.

You Ouyang, Wenjie Li, Sujian Li, and Qin Lu. 2011. Applying regression models to query-focused multi-document summarization. *Information Processing & Management*, 47(2):227–237.

Karolina Owczarzak, John M Conroy, Hoa Trang Dang, and Ani Nenkova. 2012. An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9. Association for Computational Linguistics.

Dragomir R Radev, Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proceedings of NAACL-ANLP*, pages 21–30. Association for Computational Linguistics.

Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.

Peter A Rankel, John M Conroy, Hoa Trang Dang, and Ani Nenkova. 2013. A decade of automatic content evaluation of news summaries: Reassessing the state of the art. In *Proceedings of ACL*, pages 131–136. Association for Computational Linguistics.

Dennis W Ruck, Steven K Rogers, Matthew Kabrisky, Mark E Oxley, and Bruce W Suter. 1990. The multilayer perceptron as an approximation to a bayes optimal discriminant function. *IEEE Transactions on Neural Networks*, 1(4):296–298.

Xiaojun Wan and Jianguo Xiao. 2009. Graph-based multi-modality learning for topic-focused multi-document summarization. In *Proceedings of IJCAI*, pages 1586–1591. AAAI Press.

Xiaojun Wan and Jianwu Yang. 2008. Multi-document summarization using cluster-based link analysis. In *Proceedings of SIGIR*, pages 299–306. ACM.

Xiaojun Wan and Jianmin Zhang. 2014. Ctsum: extracting more certain summaries for news articles. In *Proceedings of SIGIR*, pages 787–796. ACM.

Xiaojun Wan, Ziqiang Cao, Furu Wei, Sujian Li, and Ming Zhou. 2015. Multi-document summarization via discriminative summary reranking. *arXiv preprint arXiv:1507.02062*.

Xiaojun Wan. 2008. An exploration of document impact on graph-based multi-document summarization. In *Proceedings of EMNLP*, pages 755–762. Association for Computational Linguistics.

Xiaojun Wan. 2011. Using bilingual information for cross-language document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1546–1555. Association for Computational Linguistics.

Wen-tau Yih, Joshua Goodman, Lucy Vanderwende, and Hisami Suzuki. 2007. Multi-document summarization by maximizing informative content-words. In *Proceedings of IJCAI*, volume 7, pages 1776–1782. AAAI Press.