Learning Summary Prior Representation for Extractive Summarization

Ziqiang Cao[1,2*] **Furu Wei**[3] **Sujian Li**[1,2] **Wenjie Li**[4] **Ming Zhou**[3] **Houfeng Wang**[1,2]
[1]Key Laboratory of Computational Linguistics, Peking University, MOE, China
[2]Collaborative Innovation Center for Language Ability, Xuzhou, Jiangsu, China
[3]Microsoft Research, Beijing, China
[4]Computing Department, Hong Kong Polytechnic University, Hong Kong
{ziqiangyeah, lisujian, wanghf}@pku.edu.cn
{furu, mingzhou}@microsoft.com cswjli@comp.polyu.edu.hk

Abstract

In this paper, we propose the concept of summary prior to define how much a sentence is appropriate to be selected into summary without consideration of its context. Different from previous work using manually compiled document-independent features, we develop a novel summary system called PriorSum, which applies the enhanced convolutional neural networks to capture the summary prior features derived from length-variable phrases. Under a regression framework, the learned prior features are concatenated with document-dependent features for sentence ranking. Experiments on the DUC generic summarization benchmarks show that PriorSum can discover different aspects supporting the summary prior and outperform state-of-the-art baselines.

1 Introduction

Sentence ranking, the vital part of extractive summarization, has been extensively investigated. Regardless of ranking models (Osborne, 2002; Galley, 2006; Conroy et al., 2004; Li et al., 2007), feature engineering largely determines the final summarization performance. Features often fall into two types: document-dependent features (e.g., term frequency or position) and document-independent features (e.g., stopword ratio or word polarity). The latter type of features take effects due to the fact that, a sentence can often be judged by itself whether it is appropriate to be included in a summary no matter which document it lies in. Take the following two sentences as an example:

1. Hurricane Emily slammed into Dominica on September 22, causing 3 deaths with its wind gusts up to 110 mph.
2. It was Emily, the hurricane which caused 3 deaths and armed with wind guests up to 110 mph, that slammed into Dominica on Tuesday.

The first sentence describes the major information of a hurricane. With similar meaning, the second sentence uses an emphatic structure and is somewhat verbose. Obviously the first one should be preferred for a news summary. In this paper, we call such fact as summary prior nature[1] and learn document-independent features to reflect it.

In previous summarization systems, though not well-studied, some widely-used sentence ranking features such as the length and the ratio of stopwords, can be seen as attempts to measure the summary prior nature to a certain extent. Notably, Hong and Nenkova (2014) built a state-of-the-art summarization system through making use of advanced document-independent features. However, these document-independent features are usually hand-crafted, difficult to exhaust each aspect of the summary prior nature. Meanwhile, items representing the same feature may contribute differently to a summary. For example, "September 22" and "Tuesday" are both indicators of time, but the latter seldom occurs in a summary due to uncertainty. In addition, to the best of our knowledge, document-independent features beyond word level (e.g., phrases) are seldom involved in current research.

The CTSUM system developed by Wan and Zhang (2014) is the most relevant to ours. It attempted to explore a context-free measure named certainty which is critical to ranking sentences in summarization. To calculate the certainty score, four dictionaries are manually built as features and a corpus is annotated to train the feature weights using Support Vector Regression (SVR). How-

[*]Contribution during internship at Microsoft Research

[1]In this paper, "summary prior features" and "document-independent features" hold the same meaning.

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), pages 829–833, Beijing, China, July 26-31, 2015.

ever, a low certainty score does not always represent low quality of being a summary sentence. For example, the sentence below is from a topic about "Korea nuclear issue" in DUC 2004: *Clinton acknowledged that U.S. is not yet certain that the suspicious underground construction project in North Korea is nuclear related.* The underlined phrases greatly reduce the certainty of this sentence according to Wan and Zhang (2014)'s model. But, in fact, this sentence can summarize the government's attitude and is salient enough in the related documents. Thus, in our opinion, certainty can just be viewed as a specific aspect of the summary prior nature.

To this end, we develop a novel summarization system called PriorSum to automatically exploit all possible semantic aspects latent in the summary prior nature. Since the Convolutional Neural Networks (CNNs) have shown promising progress in latent feature representation (Yih et al., 2014; Shen et al., 2014; Zeng et al., 2014), PriorSum applies CNNs with multiple filters to capture a comprehensive set of document-independent features derived from length-variable phrases. Then we adopt a two-stage max-over-time pooling operation to associate these filters since phrases with different lengths may express the same aspect of summary prior. PriorSum generates the document-independent features, and concatenates them with document-dependent ones to work for sentence regression (Section 2.1).

We conduct extensive experiments on the DUC 2001, 2002 and 2004 generic multi-document summarization datasets. The experimental results demonstrate that our model outperforms state-of-the-art extractive summarization approaches. Meanwhile, we analyze the different aspects supporting the summary prior in Section 3.3.

2 Methodology

Our summarization system PriorSum follows the traditional extractive framework (Carbonell and Goldstein, 1998; Li et al., 2007). Specifically, the sentence ranking process scores and ranks the sentences from documents, and then the sentence selection process chooses the top ranked sentences to generate the final summary in accordance with the length constraint and redundancy among the selected sentences.

Sentence ranking aims to measure the saliency score of a sentence with consideration of both document-dependent and document-independent features. In this study, we apply an enhanced version of convolutional neural networks to automatically generate document-independent features according to the summary prior nature. Meanwhile, some document-dependent features are extracted. These two types of features are combined in the sentence regression step.

2.1 Sentence Ranking

PriorSum improves the standard convolutional neural networks (CNNs) to learn the summary prior since CNN is able to learn compressed representation of n-grams effectively and tackle sentences with variable lengths naturally. We first introduce the standard CNNs, based on which we design our improved CNNs for obtaining document-independent features.

The standard CNNs contain a convolution operation over several word embeddings, followed by a pooling operation. Let $v_i \in \mathbb{R}^k$ denote the k-dimensional word embedding of the i_{th} word in the sentence. Assume $v_{i:i+j}$ to be the concatenation of word embeddings $v_i, \cdots, v_{i+j}$. A convolution operation involves a filter $\mathbf{W}_t^h \in \mathbb{R}^{l \times hk}$, which operates on a window of h words to produce a new feature with l dimensions:

$$c_i^h = f(\mathbf{W}_t^h \times v_{i:i+h-1}) \quad (1)$$

where f is a non-linear function and $tanh$ is used like common practice. Here, the bias term is ignored for simplicity. Then $\mathbf{W}_t^h$ is applied to each possible window of h words in the sentence of length N to produce a feature map: $\mathbf{C}^h = [c_1^h, \cdots, c_{N-h+1}^h]$. Next, we adopt the widely-used max-over-time pooling operation (Collobert et al., 2011) to obtain the final features $\hat{c}^h$ from $\mathbf{C}^h$. That is, $\hat{c}^h = \max\{\mathbf{C}^h\}$. The idea behind this pooling operation is to capture the most important features in a feature map.

In the standard CNNs, only the fixed-length windows of words are considered to represent a sentence. As we know, the variable-length phrases composed of a sentence can better express the sentence and disclose its summary prior nature. To make full use of the phrase information, we design an improved version of the standard CNNs, which use multiple filters for different window sizes as well as two max-over-time pooling operations to get the final summary prior representation. Specifically, let $\mathbf{W}_t^1, \cdots, \mathbf{W}_t^m$ be m filters for window

sizes from 1 to m, and correspondingly we can obtain m feature maps $\mathbf{C}^1, \cdots, \mathbf{C}^m$. For each feature map $\mathbf{C}^i$, We first adopt a max-over-time pooling operation $max\{\mathbf{C}^i\}$ with the goal of capturing the most salient features from each window size i. Next, a second max-over-time pooling operation is operated on all the windows to acquire the most representative features. To formulate, the document independent features x_p can be generated by:

$$x_p = \max\{\max\{\mathbf{C}^1\}, \cdots, \max\{\mathbf{C}^m\}\}. \quad (2)$$

Kim (2014) also uses filters with varying window sizes for sentence-level classification tasks. However, he reserves all the representations generated by filters to a fully connected output layer. This practice greatly enlarges following parameters and ignores the relation among phrases with different lengths. Hence we use the two-stage max-over-time pooling to associate all these filters.

Besides the features x_p obtained through the CNNs, we also extract several document-dependent features notated as x_e, shown in Table 1. In the end, x_p is combined with x_e to conduct sentence ranking. Here we follow the regression framework of Li et al. (2007). The sentence saliency y is scored by ROUGE-2 (Lin, 2004) (stopwords removed) and the model tries to estimate this saliency.

$$\phi = [x_p, x_e] \quad (3)$$

$$\hat{y} = w_r^T \times \phi \quad (4)$$

where $w_r \in R^{l+|x_e|}$ is the regression weights. We use linear transformation since it is convenient to compare with regression baselines (see Section 3.2).

Feature	Description
POSITION	The position of the sentence.
AVG-TF	The averaged term frequency values of words in the sentence.
AVG-CF	The averaged cluster frequency values of words in the sentence.

Table 1: Extracted document-dependent features.

2.2 Sentence Selection

A summary is obliged to offer both informative and non-redundant content. Here, we employ a simple greedy algorithm to select sentences, similar to the MMR strategy (Carbonell and Goldstein, 1998). Firstly, we remove sentences less than 8 words (as in Erkan and Radev (2004)) and sort the rest in descending order according to the estimated saliency scores. Then, we iteratively dequeue one sentence, and append it to the current summary if it is non-redundant. A sentence is considered non-redundant if it contains more new words compared to the current summary content. We empirically set the cut-off of new word ratio to 0.5.

3 Experiments

3.1 Experiment Setup

In our work, we focus on the generic multi-document summarization task and carry out experiments on DUC 2001 2004 datasets. All the documents are from newswires and grouped into various thematic clusters. The summary length is limited to 100 words (665 bytes for DUC 2004). We use DUC 2003 data as the development set and conduct a 3-fold cross-validation on DUC 2001, 2002 and 2004 datasets with two years of data as training set and one year of data as test set.

We directly use the look-up table of 25-dimensional word embeddings trained by the model of Collobert et al. (2011). These small word embeddings largely reduces model parameters. The dimension l of the hidden document-independent features is experimented in the range of $[1, 40]$, and the window sizes are experimented between 1 and 5. Through parameter experiments on development set, we set $l = 20$ and $m = 3$ for PriorSum. To update the weights W_t^h and w_r, we apply the diagonal variant of AdaGrad with mini-batches (Duchi et al., 2011).

For evaluation, we adopt the widely-used automatic evaluation metric ROUGE (Lin, 2004), and take ROUGE-1 and ROUGE-2 as the main measures.

3.2 Comparison with Baseline Methods

To evaluate the summarization performance of PriorSum, we compare it with the best peer systems (PeerT, Peer26 and Peer65 in Table 2) participating DUC evaluations. We also choose as baselines those state-of-the-art summarization results on DUC (2001, 2002, and 2004) data. To our knowledge, the best reported results on DUC 2001, 2002 and 2004 are from R2N2 (Cao et al., 2015), ClusterCMRW (Wan and Yang, 2008) and REGSUM[2] (Hong and Nenkova, 2014) respectively. R2N2 applies recursive neural networks to learn

[2]REGSUM truncates a summary to 100 words.

feature combination. ClusterCMRW incorporates the cluster-level information into the graph-based ranking algorithm. REGSUM is a word regression approach based on some advanced features such as word polarities (Wiebe et al., 2005) and categories (Tausczik and Pennebaker, 2010). For these three systems, we directly cite their published results, marked with the sign "*" as in Table 2. Meanwhile, LexRank (Erkan and Radev, 2004), a commonly-used graph-based summarization model, is introduced as an extra baseline. Comparing with this baseline can demonstrate the performance level of regression approaches. The baseline StandardCNN means that we adopt the standard CNNS with fixed window size for summary prior representation.

To explore the effects of the learned summary prior representations, we design a baseline system named **Reg_Manual** which adopts manually-compiled document-independent features such as NUMBER (whether number exist), NENTITY (whether named entities exist) and STOPRATIO (the ratio of stopwords). Then we combine these features with document-dependent features in Table 1 and tune the feature weights through LIBLINEAR[3] support vector regression.

From Table 2, we can see that PriorSum can achieve a comparable performance to the state-of-the-art summarization systems R2N2, ClusterCMRW and REGSUM. With respect to baselines, PriorSum significantly[4] outperforms Reg_Manual which uses manually compiled features and the graph-based summarization system LexRank. Meanwhile, PriorSum always enjoys a reasonable increase over StandardCNN, which verifies the effects of the enhanced CNNs. It is noted that StandardCNN can also achieve the state-of-the-art performance, indicating the summary prior representation really works.

3.3 Analysis

In this section, we explore what PriorSum learns according to the summary prior representations. Since the convolution layer follows a linear regression output, we apply a simple strategy to measure how much the learned document-independent features contribute to the saliency estimation. Specifically, for each sentence, we ignore its document-dependent features through setting their values as zeros and then apply a linear transformation using the weight w_r to get a summary prior score x_p. The greater the score, the more possible a sentence is to be included in a summary without context consideration. We analyze what intuitive features are hidden in the summary prior representation.

[3] http://www.csie.ntu.edu.tw/~cjlin/liblinear/

[4] T-test with p-value ≤ 0.05

Year	System	ROUGE-1	ROUGE-2
2001	PeerT	33.03	7.86
	R2N2*	35.88	7.64
	LexRank	33.43	6.09
	Reg_Manual	34.55	7.18
	StandardCNN	35.19	7.63
	PriorSum	**35.98**	**7.89**
2002	Peer26	35.15	7.64
	ClusterCMRW*	**38.55**	8.65
	LexRank	35.29	7.54
	Reg_Manual	34.81	8.12
	StandardCNN	35.73	8.69
	PriorSum	36.63	**8.97**
2004	Peer65	37.88	9.18
	REGSUM*	38.57	9.75
	LexRank	37.87	8.88
	Reg_Manual	37.05	9.34
	StandardCNN	37.90	9.93
	PriorSum	**38.91**	**10.07**

Table 2: Comparison results (%) on DUC datasets.

high scored	Meanwhile, Yugoslavia's P.M. told an emergency session Monday that the country is faced with war.
	The rebels ethnic Tutsis, disenchanted members of President Laurent Kabila's army took up arms, creating division among Congo's 400 tribes.
	The blast killed two assailants, wounded 21 Israelis and prompted Israel to suspend implementation of the peace accord with the Palestinians.
low scored	The greatest need is that many, many of us have been psychologically traumatized, and very, very few are receiving help.
	Ruben Rivera: An impatient hitter who will chase pitches out of the strike zone.
	I think we should worry about tuberculosis and the risk to the general population.

Table 3: Example sentences selected by prior scores.

From Table 3, first we find that high-scored sentences contains more named entities and numbers, which conforms to human intuition. By contrast, the features NENTITY and NUMBER in Reg_Manual hold very small weights, only $2\%, 3\%$ compared with the most significant feature AVG-CF. One possible reason is that named entities or numbers are not independent features. For example, "month + number" is a common timestamp for an event whereas "number + a.m." is over-detailed and seldom appears in a summary. We can also see that low-scored sentences are relatively informal and fail to provide facts, which

are difficult for human to generalize some specific features. For instance, informal sentences seem to have more stopwords but the feature STOPRATIO holds a relatively large positive weight in Reg_Manual.

4 Conclusion and Future Work

This paper proposes a novel summarization system called PriorSum to automatically learn summary prior features for extractive summarization. Experiments on the DUC generic multi-document summarization task show that our proposed method outperforms state-of-the-art approaches. In addition, we demonstrate the dominant sentences discovered by PriorSum, and the results verify that our model can learn different aspects of summary prior.

Acknowledgments

We thank all the anonymous reviewers for their insightful comments. This work was partially supported by National Key Basic Research Program of China (No. 2014CB340504), National Natural Science Foundation of China (No. 61273278 and 61272291), and National Social Science Foundation of China (No: 12&ZD227). The correspondence author of this paper is Sujian Li.

References

Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. 2015. Ranking with recursive neural networks and its application to multi-document summarization. In *AAAI-2015*.

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.

Ronan Collobert, Jason Weston, Lon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

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

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.(JAIR)*, 22(1):457–479.

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

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

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

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

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the ACL Workshop*, pages 74–81.

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

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning semantic representations using convolutional neural networks for web search. In *Companion publication of the 23rd international conference on World wide web companion*, pages 373–374.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54.

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

Xiaojun Wan and Jianmin Zhang. 2014. Ctsum: extracting more certain summaries for news articles. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 787–796. ACM.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210.

Wen-tau Yih, Xiaodong He, and Christopher Meek. 2014. Semantic parsing for single-relation question answering. In *Proceedings of ACL*.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING*, pages 2335–2344.