Proceedings of Machine Learning Research 157, 2021 ACML 2021

Hybrid Summarization with Semantic Weighting Reward and Latent Structure Detector

Mingyang Song MINGYANG.SONG@BJTU.EDU.CN
Liping Jing* LPJING@BJTU.EDU.CN
Yi Feng 17281238@BJTU.EDU.CN
Zhiwei Sun 19120401@BJTU.EDU.CN
Lin Xiao 17112079@BJTU.EDU.CN
Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, China

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

Text summarization has been a significant challenge in the Nature Process Language (NLP) field. The approach of dealing with text summarization can be roughly divided into two main paradigms: extractive and abstractive manner. The former allows capturing the most representative snippets in a document while the latter generates a summary by understanding the latent meaning in a material with a language generation model. Recently, studies found that jointly employing the extractive and abstractive summarization models can take advantage of their complementary advantages, creating both concise and informative summaries. However, the reinforced summarization models mainly depend on the ROUGE-based reward, which only has the ability to quantify the extent of word-matching rather than semantic-matching between document and summary. Meanwhile, documents are usually collected with redundant or noisy information due to the existence of repeated or irrelevant information in real-world applications. Therefore, only depending on ROUGE-based reward to optimize the reinforced summarization models may lead to biased summary generation. In this paper, we propose a novel deep **Hy**brid **S**ummarization with semantic weighting **R**eward and latent structure **D**etector (HySRD). Specifically, HySRD introduces a new reward mechanism that simultaneously takes advantage of semantic and syntactic information among documents and summaries. To effectively model the accuracy semantics, a latent structure detector is designed to incorporate the high-level latent structures in the sentence representation for information selection. Extensive experiments have been conducted on two well-known benchmark datasets *CNN/Daily Mail* (short input document) and *BigPatent* (long input document). The automatic evaluation shows that our approach significantly outperforms the state-of-the-art of hybrid summarization models.

Keywords: Text Summarization, Reinforcement Learning, Representation Learning

1. Introduction

Neural document summarization aims to condense the given document and generate a concise version with salient information, which has attracted increasing attentions in the field of natural language processing and machine learning. However, it is still a challenging task because the collected documents (especially long document) usually contain redundant or noisy information. The main issue focuses on how to filter out the redundant information and select salient content from the given document for generating a summary.

* Corresponding author.

There are two prominent types of summarization algorithms. First, extractive summarization methods form summaries by copying parts of the input Nallapati et al. (2016b, 2017). Second, abstractive summarization systems generate new phrases, possibly rephrasing or using words that were not in the original text Rush et al. (2015); Nallapati et al. (2016a).

Inspired by the attention encoder-decoder model Bahdanau et al. (2015), recent neural document summarization approaches have been proposed to select the important parts from documents. The typical approaches include hierarchical attention Nallapati et al. (2016a), copy-pointer See et al. (2017), mixture model Hsu et al. (2018) and etc. Unfortunately, these approaches are followed by minimizing the cross-entropy loss or the maximum-likelihood loss, which makes them suffer from the exposure bias Ranzato et al. (2016) especially when handling very long documents (contains more redundant or noisy information).

One way to remedy this is to learn a policy that maximizes a specific discrete metric (i.e., ROUGE evaluation metric) instead of minimizing the cross-entropy loss or the maximum-likelihood loss, which is made possible with reinforcement learning. Thus, Paulus et al. (2018) use the self-critical policy gradient training algorithm Rennie et al. (2017) to optimize the ROUGE-based reward for the abstractive summarization model. Chen and Bansal (2018) pro- posed to apply policy gradient methods with rewards from sentence-level ROUGE to train an extractor agent. Later, Narayan et al. (2018) proposed policy gradient with rewards from summary-level ROUGE. They defined an action as sampling a summary from candidate summaries that contain the limited number of plausible sentences. After training, a sentence is ranked high for selection if it often occurs in high-scoring summaries. However, a good summary should maintain the saliency, directed logical entailment, and non-redundancy simultaneously. Pasunuru and Bansal (2018) address these three important aspects of a good summary via a reinforcement learning approach with a multi-reward function. Even though summarization benefits from reinforcement learning, these two methods directly take ROUGE as the reward. As we known, ROUGE only quantifies the extent of word-matching between document and summary, which may ignore the semantic relation among them.

In the last decade, researchers made much effort to represent document. Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) are two typical models, which are widely used for different Natural Language Process (NLP) tasks. Especially, they are improved with the aid of various attention mechanisms for document summarization, such as leverage CNN and Bi-LSTM Chen and Bansal (2018). Even though these deterministic models have made significant contributions to summarization, they are limited to adequately capture the high-level concepts which are useful to identify the semantic relation between document and summary Li et al. (2017).

Motivated by the above issues, in this paper, we propose a **H**ybrid **S**ummarization method with semantic weighting **R**eward and latent structure **D**etector (HySRD). Specifically, HySRD takes advantage of syntax-based matching score and semantic-based similarity score to shape a new reward function, which has ability to supervise the agent selects salient sentences by considering both syntax and semantic information. A good by-product of the proposed reward is to reduce the variance of reward estimation (which significantly interferes the ROUGE-based reinforced summarization) in reinforcement learning algorithm, thus, it is expected to output a good summary. Meanwhile, in order to effectively capture the semantic information with less redundant or noisy, a new information distillation model is presented by introducing neural variational auto-encoders. The sentences representation are filtered by exploiting the high-level structures of the global (document) information. To make information filtering and summarization generation enhance each other, they are seamlessly integrated in an end-to-end reinforcement learning architecture, which can be efficiently implemented via a joint optimizing algorithm. The experimental results on benchmark datasets (*CNN/Daily Mail* and *BigPatent*) have shown that HySRD outperforms the state-of-the-art baselines.

To summarize, our contributions are threefold:

- HySRD incorporates the latent structure detector into the hybrid reinforced summarization framework for filtering out the secondary information from the representation of the sentence and capturing the accuracy semantics of sentences.
- A new semantic weighting reward mechanism is presented to select the salient sentences for reinforced summarization, which simultaneously considers word-level syntactic matching and high-level semantic matching among documents and summaries.
- Experimental results indicate that our methods significantly outperform the state-of-the-art summarization models on two benchmark datasets CNN/DailyMail and BigPatent.

2. Background

2.1. Hybrid Summarization

In this paper, we focus on single-document multi-sentence summarization and propose a new model based on the hybrid summarization framework. It firstly uses an extractive agent as a sentence extractor to select the important sentences from the given document, and then leverages an abstractive module to rewrite and compress the extracted sentences.

2.2. Learning Sentence Selection

The most typical approach to train an extractor network to select informative sentences is building extractive oracles as gold targets via some heuristics (i.e., ROUGE), and training with cross-entropy loss. An oracle consists of a set of sentences with the highest possible ROUGE scores. Building oracles is finding an optimal combination of the article sentences, where there are 2^n possible combinations for each example. Because of this, the exact optimization for ROUGE scores is intractable. Therefore, alternative methods identify the set of sentences with greedy strategy Nallapati et al. (2017), sentence search Hsu et al. (2018) or two-step extraction Zhong et al. (2020), which construct suboptimal oracles. Even if all the optimal oracles are found, training with cross-entropy loss using these labels will cause underfitting as it will only maximize probabilities for sentences in label sets and ignore all other sentences Bae et al. (2019). Alternatively, reinforcement learning can give room for exploration in the search space. Xiao et al. (2020), our backbone, proposed to apply policy gradient methods to train an extractor. This approach makes an end-to-end trainable stochastic computation graph, encouraging the model to select sentences with high ROUGE scores.

3. Preliminary

Given a document-summary pair $\{\mathbf{x}, \mathbf{y}\}$, and $\mathbf{x}$ and $\mathbf{y}$ are the document and reference summary training pair. For a long text document $\mathbf{x}$ with a sequence $(x_1, ..., x_n, ...x_N)$ containing N sentences, summarization aims to output a readable multi-sentence summary. Each sentence x_n is made up of a sequence of M words $(w_{n,1}, ..., w_{n,m}, ...w_{n,M})$. For trainining data, each document has the corresponding supervised information, i.e., a sequence of J sentences $\mathbf{y} = (y_1, ..., y_j, ...y_J)$, to form the ground truth summary. The goal of summarization task, for a given document, is to predict the summary ($\hat{\mathbf{y}} = (\hat{y}_1, ..., \hat{y}_j, ...\hat{y}_J)$), so that the prediction $\hat{\mathbf{y}}$ approaches to $\mathbf{y}$ as much as possible.

For document summarization task, a feasible way is to combine the extractor with the abstractor through a specific strategy to form an end-to-end process. To reach this goal, a successful framework is hybrid extractive-abstractive architecture with policy-based reinforcement learning Xiao et al. (2020). At the t-th step, the extractor receives a document $\mathbf{x}$ as state s_t and extracts its one sentence as an action a_t. The optimal action is selected according to a policy π_θ (a mapping from states to actions). In this case, the extractor agent can be taken as a stochastic policy π_{θ_a} (with parameters θ_a) to select actions (sentences) according to the corresponding ROUGE scores for reinforced summarization. Specially, the reward of the t-th generated sentence (s'_t) can be calculated as: $r_{i,t} = \mathit{MR}(s'_t, \mathbf{y})$, where *MR* indicates the Marginal Reward Xiao et al. (2020) between the generated sentence (s'_t) and the reference summary ($\mathbf{y}$), and s'_t is obtained by rewriting the extracted sentence d_t. At the current step t, the total discounted future reward can be defined as:

$$R_t(s'_t) = \sum_{\tau=0}^{N-t} \gamma^\tau r_{t+\tau}. \tag{1}$$

Here γ is the discounted factor. To learn the optimal policy π_θ, *SentenceRewriting* exploits a critic network to predict a *baseline* b_t, which is used to estimate the advantage value for each action, $A_t(s'_t) = R_t(s'_t) - b_t$. The critic network can be trained by minimizing the square loss, $L_{\theta_c} = (b_t - R_t(s'_t))^2$. The final goal is to maximize the advantage value along all actions for the extractor agent:

$$L_{rl} = -\frac{1}{N_s}\sum_{t=1}^{N_s}[\log\pi_{\theta_a} A_t(s'_t)], \tag{2}$$

where N_s is the number of extracted sentences. This kind of hybrid model takes advantages of both extractive method and abstractive method via policy-based reinforcement learning, so that a readable summary can be generated efficiently.

4. The Proposed Method

From the preliminary, it can be seen that the optimal neural summarization model depends on the advantage value ($A_t^{\pi_\theta}(s_t, a_t)$) at each step. In this section, we present a semantic weighting reward strategy to enhance the semantical consistency between the generated summarization and the original document. To sufficiently capture the document information, following the reward strategy, a novel hierarchical representation learning model is proposed by considering the high-level concepts of both global document and local sentences.

4.1. Latent Structure Detector

To make the learning process much more efficient, both document and summary are represented in sentence-level, which has been proven powerful to evaluate the sentence saliency and further improve summarization performance. Meanwhile, the abstractive summarization models aim to create new sentences, which requires a deeper understanding of the document, such as the inherent structures “What-Happened” and “Who Action What” in CNN data Li et al. (2017). Intuitively, incorporating latent structure information into the abstractive summarization model will improve

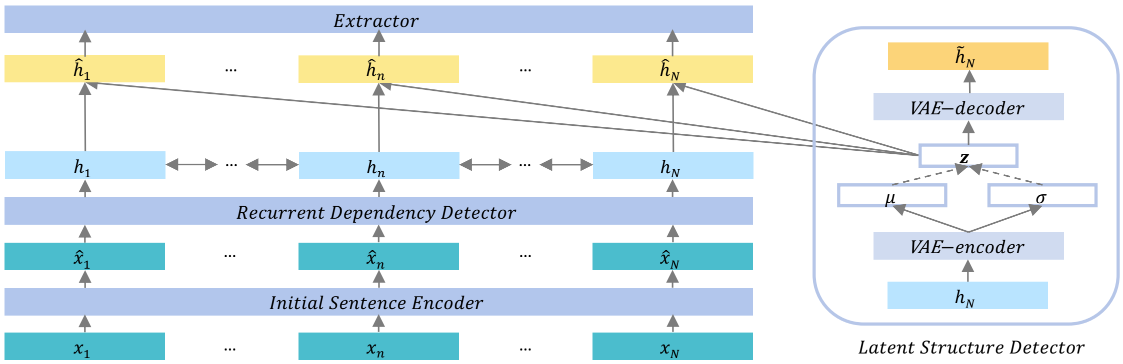

Figure 1: The proposed hierarchical architecture for sentence representation learning. It includes three parts, the first part for collecting the initial sentence information via initial sentence encoding network, the second part for capturing the recurrent dependences among sentences with a Bi-LSTM layer, and the last part to capture the latent structure information via a LSD module.

the accuracy of the extracted summaries. However, most existing summarization methods ignore this point because they adopted the discriminative sequence-to-sequence model Hsu et al. (2018); Chen and Bansal (2018); Gui et al. (2019); Moroshko et al. (2019); Sharma et al. (2019a), which limits the representation ability on the latent structure information Miao and Blunsom (2016). Thus, in this subsection, we propose a new high-level structure-aware encoding network to capture the latent structure of document among sentences.

The representation model consists of three parts. The first part aims to obtain the initial sentence information from word vectors. Specifically, the m-th word in the n-th sentence can be pre-embeded as a vector $\hat{w}_m^{(n)} \in \mathbb{R}^{d_1}$. The global information ($x_n^p \in \mathbb{R}^{d_2}$) of the n-th sentence can be modeled by:

$$x_n^p = \mathit{relu}\big(\mathbf{W}_p(\frac{1}{M}\sum_{m=1}^{M}\hat{w}_m^n)\big), \tag{3}$$

where $\mathbf{W}_p$ is a learnable mapping matrix. Meanwhile, the local information $x_n^c \in \mathbb{R}^{d_3}$ of this sentence can be computed by the temporal convolutional approach Kim (2014). Finally, the initial representation of the n-th input sentence ($\hat{x}_n \in \mathbb{R}^{d_4}$) can be obtained as follows,

$$\hat{x}_n = \mathit{relu}\big(\mathbf{W}_g[x_n^p; x_n^c]\big), \tag{4}$$

where $\mathbf{W}_g$ is a learnable matrix to combine both local and global sentence information.

The second part tries to capture the recurrent and long-term dependencies among sentences. Thus, a Bi-LSTM layer is adopted to enhance the sentence representation as follows,

$$\mathbf{h} = f_{bl}(\hat{x}_1, ..., \hat{x}_n, ..., \hat{x}_N), \tag{5}$$

where $\mathbf{h} = \{h_n\}_{n=1}^N$ and f_{bl} denotes the operation in the Bi-LSTM layer. $h_n \in \mathbb{R}^{d_5}$ is the latent state of the n-th sentence. h_N indicates the representation of the document, and the whole documents on the training dataset can be representated via $\mathbf{H}$. Similarly, for each document, its the summary can be encoded as $\mathbf{P}$ in a d_5-dimensional latent space which is same with the latent space of document.

Our goal in the third part is to capture the latent structure hidden in the document. Therefore, we apply variation autoencoder Rezende et al. (2014) as the Latent Structure Detector (LSD) for obtaining sentences representation. Each document is encoded via a latent variable $\mathbf{z} \in \mathbb{R}^{d_3}$ which is assumed to be sampled from a standard Gaussian prior, i.e., $\mathbf{z} \sim p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I}_d)$. Such variable has ability to determine the latent structure hidden in the documents and will be useful to generate summarization Li et al. (2017). During the encoding process, $\mathbf{z}$ can be sampled via a reparameterization trick for Gaussian distribution, i.e., $\mathbf{z} \sim q(\mathbf{z}|h_N) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$. Specificially, we sample an auxiliary noise variable $\boldsymbol{\varepsilon} \sim N(0, \mathbf{I})$ and reparametrize $\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \varepsilon$, where $\odot$ denotes the element-wise multiplication. The mean vector $\boldsymbol{\mu} \in \mathbb{R}^{d_3}$ and variance vector $\boldsymbol{\sigma} \in \mathbb{R}^{d_3}$ will be inferred by a two-layer network with ReLU-activated function, i.e., $\boldsymbol{\mu} = \mu_\phi(h_N)$ and $\boldsymbol{\sigma} = \sigma_\phi(h_N)$ where ϕ is the parameter set. During the decoding process, the document can be reconstructed by a muylti-layer network (f_k) with Tanh-activated function, i.e., $\tilde{h}_N = f_k(\mathbf{z})$.

To simultaneously minimize the reconstruction loss and penalize the discrepancy between prior distribution and posterior distribution about the latent variable $\mathbf{z}$, the VAE process can be implemented by optimizing the following objective function,

$$L_z = -\mathbb{E}_{q(\mathbf{z}|\mathbf{H})}\big[p(\mathbf{H}|\mathbf{z})\big] + D_{KL}\big(p(\mathbf{z})||q(\mathbf{z}|\mathbf{H})\big), \tag{6}$$

where D_{KL} indicates the Kullback-Leibler divergence between two distributions. Specifically, the latent structure detector is trained with the whole framework synchronously.

Once having the latent document representation $\mathbf{z}$, the sentence representation h_n can be enhanced by considering the global document structure information as follows,

$$\hat{h}_n = \textit{relu}(\mathbf{W}_f[h_n; \mathbf{z}]). \tag{7}$$

Here, $\hat{h}_n \in \mathbb{R}^{d_5}$ and $\mathbf{W}_f \in \mathbb{R}^{d_5 \times (d_5 + d_3)}$ is a learnable mapping matrix. For training data, the reference summary representation p_J can be enhanced with the same strategy into $\hat{p}_J$. $\{\hat{h}_n\}_{n=1}^N$ and $\{\hat{p}_J\}$ will be used to evaluate the semantic relation between sentences and summary as shown in Eq.(8) and Eq.(9).

4.2. Semantic Weighting Reward

Pieces of evidence (e.g., Vaswani et al. (2017)) show that attention mechanism Bahdanau et al. (2015) is very significant for natural language generation tasks including reinforced summarization. The neural intra-attention model Paulus et al. (2018) generates a readable long summary by designing intra-temporal attention and intra-decoder attention. However, the attention in this model directly attends over the words of input /output sequence, which is hard to capture the document's semantics because word is low-level feature Wang et al. (2019). Moreover, the loss function of reinforced learning only depends on the ROUGE score, which may result in poor performance because it does not explicitly cover the semantic information among document and summary.

To effectively and efficiently capture the semantic information, inspired by Chen and Bansal (2018), the input document and the reference summary are represented in sentence-level. Given the sequence of sentences, the input sentence at the n-th step is represented via a hidden state $\hat{h}_n$ and the document is represented by the last hidden state $\hat{h}_N$ (details will be given in next Section). By calculating the similarity between $\hat{h}_n$ and $\hat{h}_N$, we can capture the importance of the n-th sentence in document at the semantic-level. Hence, we propose to explicitly calculate the sentence-level attention score β_n between sentence $\hat{h}_n$ and document $\hat{h}_N$ by simple scalar multiplication and renormalization as follows,

$$\beta_n(a_t) = \sigma\Big(\frac{f_g(\hat{h}_n)\mathbf{W}_d\hat{h}_N}{(\sum_{k=1}^{N} f_g(\hat{h}_k)\hat{h}_N)/N}\Big), \tag{8}$$

where f_g is the glimpse operation with the same computation as Vinyals et al. (2016), and $\mathbf{W}_d \in \mathbb{R}^{d_5 \times d_5}$ is a learnable mapping matrix. This attention mechanism makes sure that the sentence has a higer attention β_n only when the correlation between sentence and document is higher than average. Intuitively, the higher the attention value of a sentence is, the more important the sentence is and the corresponding sentence should be selected by the extractor agent for reinforced summarization.

In this paper, we focus on supervised document summarization, i.e., the training process is supervised by the ground-truth summary. Thus, it will be intuitive to modulate the semantic attention between sentences and document with the aid of ground-truth summary, so that the selected sentence is more useful to generate summary. Specifically, the reference summary can be represented by a hidden state $\hat{p}_J$ (with the same encoding network with document representation). The contribution of the n-th input sentence ($\hat{\beta}_n$) to the final summary can be quantified by:

$$\hat{\beta}_n(a_t) = \sigma\Big(\frac{f_g(\hat{h}_n)\mathbf{W}_s\hat{p}_J}{(\sum_{k=1}^{N} f_g(\hat{h}_k)\hat{p}_J)/N}\Big), \tag{9}$$

where $\mathbf{W}_s \in \mathbb{R}^{d_5 \times d_5}$ is a learnable mapping matrix and σ indicates the sigmoid function. The attention β_n and $\hat{\beta}_n$ can be aligned by the following mean square loss,

$$L_s = \frac{1}{N_s}\sum_{t=1}^{N_s}(\beta_t(a_t) - \hat{\beta}_t(a_t))^2. \tag{10}$$

This semantic attention is helpful to check the extent to which the generated sentence is semantically related to the ground-truth summary.

To take advantage of the semantic relation among document and summary, a semantic weighting reward is designed to combine word-matching ROUGE score and semantic-matching attention mechanism as $A_t^{\pi_\theta}(s_t, a_t)\beta_t(\hat{h}_N, a_t)$. Then, the loss function L_{rl} of the reinforcement learning process can be written as:

$$L_{rl} = \frac{1}{N_s}\sum_{t=1}^{N_s}\Big[\log\pi_\theta(s_t, a_t)A_t^{\pi_\theta}(s_t, a_t)\beta_t(a_t)\Big], \tag{11}$$

where β_t indicates the attention score of the t-th action a_t (the extracted sentence), and $A_t^{\pi_\theta}$ is the advantage value. This reinforcement learning objective can not only make sentence selection more accurate but also increase the performance of attention optimization, which will provide a good foundation to generate stable and satisfied summary.

5. Experimental Settings

To evaluate the proposed HySRD, a series of experiments are conducted on two well-known datasets. The experimental results are discussed by comparing with the state-of-the-art baselines. Statistical information of CNN/Daily Mail and BigPatent please refers to 1.

Table 1: Statistics of *CNN/Daily Mail* and *BigPatent* datasets. # Document: raw number of documents in each dataset. For all other columns, mean values are reported over all documents.

Dataset	# Document	Document	Reference Summary	
		# word	# sentence	# word
CNN/Daily Mail	312,085	789.9	3.8	55.6
BigPatent	1,341,362	**3572.8**	3.5	116.5

5.1. Datasets

We evaluate the proposed approach on two large-scale datasets *CNN/Daily Mail* Hermann et al. (2015) and *BigPatent* Sharma et al. (2019b), which are standard corpora for multi-sentence abstractive summarization.

CNN/Daily Mail contains news stories in *CNN* and *Daily Mail* websites. Following See et al. (2017), the non-anonymized version is adopted which has 287,226 training pairs, 13,368 validation pairs and 11,490 testing pairs. The average number of sentences in document and summaries are respectively 42.1 and 3.8. We followed the pre-processing methods in See et al. (2017) after splitting sentences by Stanford CoreNLP Manning et al. (2014).

BigPatent consists of 1,341,362 U.S. patent documents, which has 1,207,222 training pairs, 67,068 validation pairs and 67,072 test pairs. The average number of document sentences and summary sentences are 3572.8 and 116.5 respectively. *BigPatent* is much harder than *CNN/Daily Mail* because documents and summaries are much longer.

5.2. Evaluation Metrics

In order to validate the summarization performance, the well-known and widely used metric ROUGE Lin (2004) is adopted to count the number of overlapping units between the generated summaries and the reference summaries. F-measures of ROUGE-1 (R-1), ROUGE-2 (R-2) and ROUGE-L (R-L) are reported. R-AVG calculates average score of R-1, R-2 and R-L. Meanwhile, the evaluation metric METEOR Banerjee and Lavie (2005) is also adopted for a more thorough analysis. Larger ROUGE and METEOR values indicate better performance.

5.3. Hyperparameter Details

The hyperparameters of HySRD are set as follows. Each word is pre-trained and represented as a 128-dimension vector. Meanwhile, the size of sentence vector in different layers are set by $d_1 = d_2 = 128$, $d_3 = d_4 = 300$, $d_5 = 512$. For each dataset, the most frequently 30000 words are kept as the vocabulary. For optimization, Adam is used with learning rate 10^{-4}, and the mini-batches size is 32. When calculating the reward, the discount factor is set as $\gamma = 0.95$. During reference, we apply the beam search Paulus et al. (2018) with width 5 on the abstractor to avoid trigram repetition.

6. Results

We will show the experimental results on *CNN/Daily Mail* and *BigPatent* to demonstrate the superiority of HySRD over the state-of-the-art baselines.

Table 2: Comparing the summarization performance on *CNN/Daily Mail* testing dataset in terms ROUGE-1 (**R-1**), ROUGE-2 (**R-2**), ROUGE-L (**R-L**) , **R-AVG** and **METEOR**. All ROUGE scores computed by the official ROUGE script have 95% confidence interval of at most ±0.19. Here, LSD indicates the latent structure detector and SWR indicates the semantic weighting reward. A2C denotes the policy gradient algorithm on the hybrid summarization Chen and Bansal (2018). HRL denotes the hierarchical reinforcement learning on the hybrid summarization Xiao et al. (2020). ExtractorAgent + AbstractorAgent + HRL + LSD + SWR indicates the HySRD.

Model	R-1	R-2	R-L	R-AVG	METEOR
Extract-based Models					
LEAD-3 See et al. (2017)	40.34	17.70	36.57	31.54	22.21
RankingSentence Narayan et al. (2018)	40.00	18.20	36.60	31.60	-
BANDITSUM Dong et al. (2018)	41.47	18.72	37.76	32.65	22.35
Abstract-based Models					
PointerGen+Coverage See et al. (2017)	39.53	17.28	36.38	31.06	18.72
DeepRL Paulus et al. (2018)	39.87	15.82	36.90	30.86	-
MultiReward Pasunuru and Bansal (2018)	40.43	18.00	37.10	31.84	20.02
InconsistencyLoss Hsu et al. (2018)	40.68	17.97	37.13	31.93	-
SentRewriting Chen and Bansal (2018)	40.88	17.80	38.54	32.41	20.38
DCA Çelikyilmaz et al. (2018)	41.69	19.47	37.92	33.02	-
BottomUp Gehrmann et al. (2018)	41.22	18.68	38.34	32.75	-
HySum Xiao et al. (2020)	42.46	19.10	39.19	33.58	21.88
Our Results					
ExtractorAgent + Abstractor + A2C	41.15	18.37	38.88	32.80	20.62
ExtractorAgent + Abstractor + A2C + LSD	41.41	18.50	39.03	32.98	20.69
ExtractorAgent + Abstractor + A2C + SWR	41.32	18.39	39.01	32.91	20.55
ExtractorAgent + Abstractor + A2C + LSD + SWR	41.77	18.71	39.34	33.27	20.88
ExtractorAgent + AbstractorAgent + HRL	41.91	18.74	39.51	33.39	21.51
ExtractorAgent + AbstractorAgent + HRL + LSD	42.29	19.12	39.84	33.75	21.75
ExtractorAgent + AbstractorAgent + HRL + SWR	42.26	19.10	39.73	33.70	21.68
ExtractorAgent + AbstractorAgent + HRL + LSD + SWR	**42.63**	**19.41**	**39.93**	**33.99**	**21.92**

6.1. Results on Short Document (CNN/Daily Mail)

For sufficient comparison, we listed the results of HySRD obtained by its six versions: ExtractorAgent + Abstractor + A2C denotes the hybrid framework with the pre-trained abstractor module, ExtractorAgent + Abstractor + A2C + LSD increments the latent structure detector, and ExtractorAgent + Abstractor + A2C+ LSD + SWR increments the semantic weighting reward. ExtractorAgent + AbstractorAgent + HRL denotes the hybrid framework with the abstractor agent under the hierarchical reinforcement learning algorithm, ExtractorAgent + AbstractorAgent + HRL + LSD increments the latent structure detector, and ExtractorAgent + AbstractorAgent + HRL + LSD + SWR increments the semantic weighting reward. Specifically, A2C refers to the actor-critic learning algorithm Chen and Bansal (2018), and HRL refers to the hierarchical reinforcement learning algorithm Xiao et al. (2020). In this paper, we focus on reinforcement learning based two-stage

Table 3: Performance on *BigPatent* dataset using the full length ROUGE F1 score. All ROUGE scores computed by the official ROUGE script have 95% confidence interval of at most ±0.07.

Model	**R-1**	**R-2**	**R-L**	**R-AVG**
Extract-based Result				
Lead-3 See et al. (2017)	31.27	8.75	26.18	22.07
TextRank Mihalcea and Tarau (2004)	35.99	11.14	29.60	25.58
SentenceExtractRL Chen and Bansal (2018)	34.63	10.62	29.43	24.89
Abstract-based Result				
PointerGen See et al. (2017)	30.59	10.01	25.65	22.08
PointerGen + Coverage See et al. (2017)	33.14	11.63	28.55	24.44
SentenceRewriting Chen and Bansal (2018)	37.12	11.87	32.45	27.15
Our Results				
ExtractorAgent + Abstractor + A2C	37.41	12.24	32.72	27.46
ExtractorAgent + Abstractor + A2C + LSD	38.01	12.53	33.11	27.88
ExtractorAgent + Abstractor + A2C + SWR	37.92	12.49	33.05	27.82
ExtractorAgent + Abstractor + A2C + LSD + SWR	38.68	12.90	33.71	28.43
ExtractorAgent + AbstractorAgent + HRL	39.08	13.10	34.13	28.77
ExtractorAgent + AbstractorAgent + HRL + LSD	39.24	13.33	34.30	28.96
ExtractorAgent + AbstractorAgent + HRL + SWR	39.19	13.26	34.21	28.89
ExtractorAgent + AbstractorAgent + HRL + LSD + SWR	**39.63**	**14.18**	**35.06**	**29.62**

abstractive summarization. Thus, for a fair comparison, the existing two-stage abstractive and reinforcement learning based summarization methods are selected as baselines. Furthermore, due to the limited computational resource, we did not adopt pre-trained language models (i.e., BERT Devlin et al. (2019)) as our backbone.

The experimental results on *CNN/Daily Mail* dataset are shown in Table 2, with extractive models in the top block and abstractive models in the second block. For comparison, we list the performance of many recent approaches with ours. Overall, our model achieves strong improvements and the new state-of-the-art on both extractive and abstractive settings for the CNN/Daily Mail dataset Comparing ExtractorAgent + Abstractor + A2C and ExtractorAgent + AbstractorAgent + HRL models, it is found that the introduction of different reinforcement learning methods can significantly influence the results.

6.2. Results on Long Document (BigPatent)

Many existing summarization models often show poor information capture ability when facing long documents Li et al. (2017); You et al. (2019). Thus, it is urgent to test the performance of HySRD on addressing long documents. Here, we evaluate our model on the benchmark dataset BigPatent Sharma et al. (2019b) to investigate whether HySRD could achieve improvement when dealing with documents containing more sentences compared with other typical extractive models. The experimental results on *BigPatent* dataset are shown in Table 3, with extractive models in the top block and abstractive models in the second block. As can be seen in Table 3, HySRD performs better than the baselines on *BigPatent* in terms of all evaluation metrics.

Comparing these two benchmark datasets (*CNN/Daily Mail* and *BigPatent*), the improvement gains of HySRD over SentRewriting are 4.28%, 8.41%, 3.60%, 4.87% and 6.76%, 19.46%, 8.04%, 9.10% (in terms of R-1, R-2, R-L and R-AVG) respectively. Obviously, HySRD significantly improves the summarization performance on *BigPatent* which contains much longer documents than *CNN/Daily Mail*. This result further demonstrates that HySRD takes advantage of the semantic weighting reward and the latent structure detector.

6.3. Ablation Study

We also conduct some ablation studies in Table 2 to verify the effectiveness of each component. For the **latent structure detector** module, we build two ablation models, ExtractorAgent + Abstractor + A2C + LSD and ExtractorAgent + AbstractorAgent + HRL + LSD, which only use the latent structure detector under different reinforcement learning algorithm for optimization. Concretely, we design the latent structure detector to improve the capability of capturing the structure information and further encode the document comprehensively. Therefore, compared with most baseline models, ExtractorAgent + AbstractorAgent + HRL + LSD achieves better F1 scores on ROUGE-1, ROUGE-2, ROUGE-L and ROUGE-AVG evaluation metrics. Even ExtractorAgent + Abstractor + A2C + LSD obtains better results than most baselines, especially SentenceRewriting. Therefore, the proposed latent structure detector for sentences representation improves the final performance. For the **semantic weighting reward** module, as expected, the proposed HySRD is superior to the existing abstractive methods including reinforced abstractive summarization methods (DeepRL; Paulus et al. (2018), MultiReward; Pasunuru and Bansal (2018), SentRewriting Chen and Bansal (2018), and DCA Çelikyilmaz et al. (2018)) on *CNN/Daily Mail* dataset in terms of all evaluation metrics. Especially, HySRD significantly improves the ROUGE-L score. As mentioned inChen and Bansal (2018), the ROUGE-L score is much important for document summarization because the generated summary with high ROUGE-L scores is more fluent. Concretely, these results confirms that the designed the semantic weighting reward has ability to optimize the summarization model to select salient information with higher accuracy. Therefore, ExtractorAgent + Abstractor + A2C + LSD + SWR and ExtractorAgent + AbstractorAgent + HRL + LSD + SWR have ability to achieve improvement in producing summaries with salient segments. The main reason, we believe, is that the latent structure detector is helpful to sufficiently represent sentences from the global point, which further benefits the semantic weighting reward calculation.

7. Related Work

In this section, we introduce the related work from two threads: 1) the combination of extractive and abstractive summarization; 2) the usage of reinforcement learning in the summarization.

The approach of dealing with text summarization can be roughly divided into two main paradigms: extractive and abstractive manner. Extractive Summarization aims to select important sentences from a document as its summary. It is usually modeled as sentence ranking task by using the scores predicted by some classifiers Nallapati et al. (2016b, 2017). They have been extended with the aid of salient estimation Shi et al. (2019) and reinforcement learning Narayan et al. (2018). Abstractive Summarization aims to generate the summary of a document from scratch. Recently, there has been a variety of deep neural network models for abstractive document summarization Rush et al. (2015); Nallapati et al. (2016a). One of the most dominant structures is based on the neural sequence-to-sequence learning framework with attention mechanism Bahdanau et al. (2015). See et al. (2017)

introduced Pointer Generator network that implicitly combines the abstraction with the word-leve extraction, using copy mechanism. More recently, there have been several researches that have attempted to improve the performance of the abstractive summarization by explicitly combining them with extractive models, such as the use of inconsistency loss Hsu et al. (2018), and sentence extraction with abstraction Chen and Bansal (2018); Bae et al. (2019); Xiao et al. (2020).

Recently, reinforcement learning (RL) has attracted increasing attentions in the field of natural document summarization due to its superiorities on optimizing the non-differential metrics and mitigating the exposure bias. In the task of reinforced summarization, it is important to design proper reward function. In literatures, various reward functions have been proposed, such as sentence-level Chen and Bansal (2018), summary-level Bae et al. (2019), and mixture-based strategy reward function Pasunuru and Bansal (2018).

HySRD is different from these above methods from two aspects. Firstly, a latent structure detector is presented to explore the high-level concepts from the global point. Secondly, a new semantic weighting reward is designed to guide the reinforcement learning process, so that the generated summary is semantically consistent with the original document.

8. Conclusions and Future Work

In this paper, we proposed a semantic weighting reward mechanism for reinforced summarization. It has ability to effectively extract salient sentences by simultaneously considering word-level syntactic matching and high-level semantic matching among documents and summaries. A series of experiments have demonstrated the superiority of the proposed method by comparing with the state-of-the-art baselines. This work focus on single-document summarization. It will be an interesting topic to extend HySRD for multi-document summarization task.

9. Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0106800; the National Science Foundation of China under Grant 61822601 and 61773050; the Beijing Natural Science Foundation under Grant Z180006; The Fundamental Research Funds for the Central Universities (2019JBZ110).

References

Sanghwan Bae, Taeuk Kim, Jihoon Kim, and Sang goo Lee. Summary level training of sentence rewriting for abstractive summarization. *CoRR*, abs/1909.08752, 2019. URL http://dblp.uni-trier.de/db/journals/corr/corr1909.html#abs-1909-08752.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2015. URL http://dblp.uni-trier.de/db/conf/iclr/iclr2015.html#BahdanauCB14.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare R. Voss, editors, *IEEvaluation@ACL*, pages 65–72. Association for Computational Linguistics,

2005. URL http://dblp.uni-trier.de/db/conf/acl/ieevaluation2005.html#BanerjeeL05.

Yen-Chun Chen and Mohit Bansal. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1063. URL https://www.aclweb.org/anthology/P18-1063.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics, 2019. ISBN 978-1-950737-13-0. URL http://dblp.uni-trier.de/db/conf/naacl/naacl2019-1.html#DevlinCLT19.

Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. Banditsum: Extractive summarization as a contextual bandit. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *EMNLP*, pages 3739–3748. Association for Computational Linguistics, 2018. ISBN 978-1-948087-84-1. URL http://dblp.uni-trier.de/db/conf/emnlp/emnlp2018.html#DongSCHC18.

Sebastian Gehrmann, Yuntian Deng, and Alexander M. Rush. Bottom-up abstractive summarization. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *EMNLP*, pages 4098–4109. Association for Computational Linguistics, 2018. ISBN 978-1-948087-84-1. URL http://dblp.uni-trier.de/db/conf/emnlp/emnlp2018.html#GehrmannDR18.

Min Gui, Junfeng Tian, Rui Wang, and Zhenglu Yang. Attention optimization for abstractive document summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1222–1228, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1117. URL https://www.aclweb.org/anthology/D19-1117.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *NIPS*, pages 1693–1701, 2015. URL http://dblp.uni-trier.de/db/conf/nips/nips2015.html#HermannKGEKSB15.

Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 132–141, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1013. URL https://www.aclweb.org/anthology/P18-1013.

Yoon Kim. Convolutional neural networks for sentence classification. 2014. URL http://arxiv.org/abs/1408.5882. cite arxiv:1408.5882Comment: To appear in EMNLP 2014.

Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. Deep recurrent generative decoder for abstractive text summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2091–2100, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: {10.18653/v1/D17-1222}. URL https://www.aclweb.org/anthology/D17-1222.

C. Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS)*, Barcelona, Spain, July 25-26 2004.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60. The Association for Computer Linguistics, 2014. ISBN 978-1-941643-00-6. URL http://dblp.uni-trier.de/db/conf/acl/acl2014-d.html#ManningSBFBM14.

Yishu Miao and Phil Blunsom. Language as a latent variable: Discrete generative models for sentence compression. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *EMNLP*, pages 319–328. The Association for Computational Linguistics, 2016. ISBN 978-1-945626-25-8. URL http://dblp.uni-trier.de/db/conf/emnlp/emnlp2016.html#MiaoB16.

Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *EMNLP*, pages 404–411. ACL, 2004. URL http://dblp.uni-trier.de/db/conf/emnlp/emnlp2004.html#MihalceaT04.

Edward Moroshko, Guy Feigenblat, Haggai Roitman, and David Konopnicki. An editorial network for enhanced document summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 57–63, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-5407. URL https://www.aclweb.org/anthology/D19-5407.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In Yoav Goldberg and Stefan Riezler, editors, *CoNLL*, pages 280–290. ACL, 2016a. ISBN 978-1-945626-19-7. URL http://dblp.uni-trier.de/db/conf/conll/conll2016.html#NallapatiZSGX16.

Ramesh Nallapati, Bowen Zhou, and Mingbo Ma. Classify or select: Neural architectures for extractive document summarization. *CoRR*, abs/1611.04244, 2016b. URL http://dblp.uni-trier.de/db/journals/corr/corr1611.html#NallapatiZM16.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In Satinder P. Singh and Shaul Markovitch, editors, *AAAI*, pages 3075–3081. AAAI Press, 2017. URL http://dblp.uni-trier.de/db/conf/aaai/aaai2017.html#NallapatiZZ17.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Ranking sentences for extractive summarization with reinforcement learning. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *NAACL-HLT*, pages 1747–1759. Association for Computational Linguistics, 2018. ISBN 978-1-948087-27-8. URL http://dblp.uni-trier.de/db/conf/naacl/naacl2018-1.html#NarayanCL18.

Ramakanth Pasunuru and Mohit Bansal. Multi-reward reinforced summarization with saliency and entailment. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *NAACL-HLT (2)*, pages 646–653. Association for Computational Linguistics, 2018. ISBN 978-1-948087-29-2. URL http://dblp.uni-trier.de/db/conf/naacl/naacl2018-2.html#PasunuruB18.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. In *ICLR (Poster)*. OpenReview.net, 2018. URL http://dblp.uni-trier.de/db/conf/iclr/iclr2018.html#PaulusXS18.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. In Yoshua Bengio and Yann LeCun, editors, *ICLR (Poster)*, 2016. URL http://dblp.uni-trier.de/db/conf/iclr/iclr2016.html#RanzatoCAZ15.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *CVPR*, pages 1179–1195. IEEE Computer Society, 2017. ISBN 978-1-5386-0457-1. URL http://dblp.uni-trier.de/db/conf/cvpr/cvpr2017.html#RennieMMRG17.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. 2014. URL http://arxiv.org/abs/1401.4082. cite arxiv:1401.4082Comment: Appears In Proceedings of the 31st International Conference on Machine Learning (ICML), JMLR: W&CP volume 32, 2014.

Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *EMNLP*, pages 379–389. The Association for Computational Linguistics, 2015. ISBN 978-1-941643-32-7. URL http://dblp.uni-trier.de/db/conf/emnlp/emnlp2015.html#RushCW15.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In Regina Barzilay and Min-Yen Kan, editors, *ACL (1)*, pages 1073–1083. Association for Computational Linguistics, 2017. ISBN 978-1-945626-75-3. URL http://dblp.uni-trier.de/db/conf/acl/acl2017-1.html#SeeLM17.

Eva Sharma, Luyang Huang, Zhe Hu, and Lu Wang. An entity-driven framework for abstractive summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3271–3282, Hong Kong, China, November 2019a. Association for Computational Linguistics. doi: 10.18653/v1/D19-1323. URL https://www.aclweb.org/anthology/D19-1323.

Eva Sharma, Chen Li, and Lu Wang. Bigpatent: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy, July 2019b. Association for Computational Linguistics. doi: 10.18653/v1/P19-1212.

Jiaxin Shi, Chen Liang, Lei Hou, Juanzi Li, Zhiyuan Liu, and Hanwang Zhang. Deepchannel: Salience estimation by contrastive learning for extractive document summarization. In *AAAI*, pages 6999–7006. AAAI Press, 2019. ISBN 978-1-57735-809-1. URL http://dblp.uni-trier.de/db/conf/aaai/aaai2019.html#ShiLHL0Z19.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *NIPS*, pages 5998–6008, 2017. URL http://dblp.uni-trier.de/db/conf/nips/nips2017.html#VaswaniSPUJGKP17.

Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. Order matters: Sequence to sequence for sets. In Yoshua Bengio and Yann LeCun, editors, *ICLR (Poster)*, 2016. URL http://dblp.uni-trier.de/db/conf/iclr/iclr2016.html#VinyalsBK15.

Yue Wang, Jing Li, Hou Pong Chan, Irwin King, Michael R. Lyu, and Shuming Shi. Topic-aware neural keyphrase generation for social media language. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *ACL (1)*, pages 2516–2526. Association for Computational Linguistics, 2019. ISBN 978-1-950737-48-2. URL http://dblp.uni-trier.de/db/conf/acl/acl2019-1.html#WangLCKLS19.

Liqiang Xiao, Lu Wang, Hao He, and Yaohui Jin. Copy or rewrite: Hybrid summarization with hierarchical reinforcement learning. In *AAAI*, pages 9306–9313. AAAI Press, 2020. ISBN 978-1-57735-823-7. URL http://dblp.uni-trier.de/db/conf/aaai/aaai2020.html#XiaoWHJ20.

Yongjian You, Weijia Jia, Tianyi Liu, and Wenmian Yang. Improving abstractive document summarization with salient information modeling. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *ACL (1)*, pages 2132–2141. Association for Computational Linguistics, 2019. ISBN 978-1-950737-48-2. URL http://dblp.uni-trier.de/db/conf/acl/acl2019-1.html#YouJLY19.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *ACL*, pages 6197–6208. Association for Computational Linguistics, 2020. ISBN 978-1-952148-25-5. URL http://dblp.uni-trier.de/db/conf/acl/acl2020.html#ZhongLCWQH20.

Asli Çelikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. Deep communicating agents for abstractive summarization. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *NAACL-HLT*, pages 1662–1675. Association for Computational Linguistics, 2018. ISBN 978-1-948087-27-8. URL http://dblp.uni-trier.de/db/conf/naacl/naacl2018-1.html#CelikyilmazBHC18.