

# Unsupervised Neural Single-Document Summarization of Reviews via Learning Latent Discourse Structure and its Ranking

Masaru Isonuma<sup>1</sup> Junichiro Mori<sup>1,2</sup> Ichiro Sakata<sup>1</sup>  
<sup>1</sup>The University of Tokyo <sup>2</sup>RIKEN  
{isonuma, isakata}@ipr-ctr.t.u-tokyo.ac.jp  
mori@mi.u-tokyo.ac.jp

## Abstract

This paper focuses on the end-to-end abstractive summarization of a single product review *without supervision*. We assume that a review can be described as a *discourse tree*, in which the summary is the root, and the child sentences explain their parent in detail. By recursively estimating a parent from its children, our model learns the latent discourse tree without an external parser and generates a concise summary. We also introduce an architecture that *ranks* the importance of each sentence on the tree to support summary generation focusing on the main review point. The experimental results demonstrate that our model is competitive with or outperforms other unsupervised approaches. In particular, for relatively long reviews, it achieves a competitive or better performance than supervised models. The induced tree shows that the child sentences provide additional information about their parent, and the generated summary abstracts the entire review.

## 1 Introduction

The need for automatic document summarization is widely increasing because of the vast amounts of online textual data that continue to grow. As for product reviews on E-commerce websites, succinct summaries allow both customers and manufacturers to obtain large numbers of opinions (Liu and Zhang, 2012). Under these circumstances, supervised neural network models have achieved wide success, using a large number of reference summaries (Wang and Ling, 2016; Ma et al., 2018). However, a model trained on these summaries cannot be adopted in other domains, as salient phrases are not common across domains. It requires a significant cost to prepare large volumes of references for each domain (Isonuma et al., 2017).

An unsupervised approach is a possible solution to such a problem. Previously, unsupervised learning has been widely applied to extractive approaches (Radev et al., 2004; Mihalcea and Tarau, 2004). As mentioned in (Carenini et al., 2013; Gerani et al., 2014), extractive approaches often fail to provide an overview of the reviews, while abstractive ones successfully condense an entire review via paraphrasing and generalization. Our work focuses on the one-sentence abstractive summarization of a single-review without supervision.

The difficulties of unsupervised abstractive summarization are two-fold: obtaining the representation of the summaries, and learning a language model to decode them. As an unsupervised approach for multiple reviews, Chu and Liu (2018) regarded the mean of the document embeddings as the summary, while learning a language model via the reconstruction of each review. By contrast, such an approach cannot be extended to a single-review directly, because it also condenses including trivial or redundant sentences (its performance is demonstrated in Section 4.4).

To overcome these problems, we apply the *discourse tree* framework. Extractive summarization and document classification techniques sometimes use a discourse parser to gain a concise representation of documents (Hirao et al., 2013; Bhatia et al., 2015; Ji and Smith, 2017); however, Ji and Smith (2017) pointed out the limitations of using external discourse parsers. In this context, Liu and Lapata (2018) proposed a framework to induce a latent discourse tree without a parser. While their model constructed the tree via a supervised document classification task, our model induces it by identifying and reconstructing a parent sentence from its children. Consequently, we gain the representation of a summary as the root of the induced latent discourse tree, while learning a language model through reconstruction.

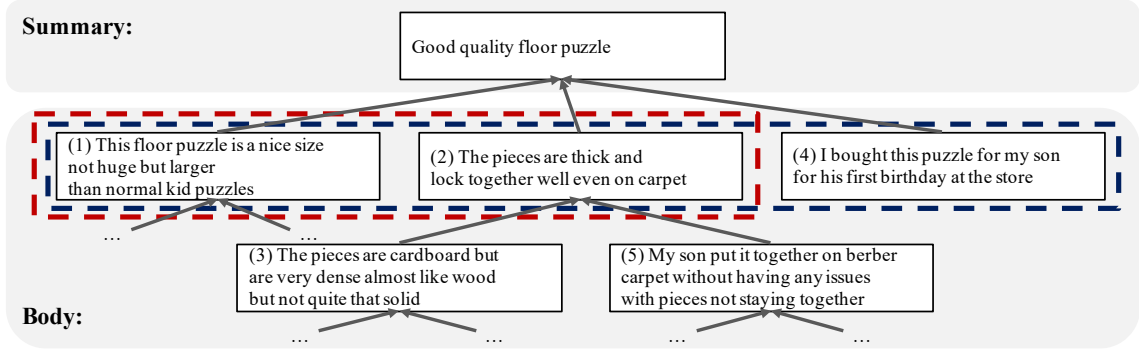


Figure 1: Example of the discourse tree of a jigsaw puzzle review. **StrSum** induces the latent tree and generates the summary from the children of a root, while **DiscourseRank** supports it to focus on the main review point.

Figure 1 shows an example of a jigsaw puzzle review and its dependency-based discourse tree. The summary describes its quality. The child sentences provide an explanation in terms of the size (1st) and thickness (2nd), or provide the background (4th). Thus, we assume that reviews can generally be described as a multi-root non-projective discourse tree, in which the summary is the root, and the sentences construct each node. The child sentences present additional information about the parent sentence.

To construct the tree and generate the summary, we propose a novel architecture; *StrSum*. It reconstructs a parent from its children recursively and induces a latent discourse tree without a parser. As a result, our model generates a summary from the *surrounding sentences of the root* while learning a language model through reconstruction in an end-to-end manner. We also introduce *DiscourseRank*, which ranks the importance of each sentence in terms of the number of *descendants*. It supports *StrSum* to generate a summary that focuses on the main review point.

The contributions of this work are three-fold:

- We propose a novel unsupervised end-to-end model to generate an abstractive summary of a single product review while inducing a latent discourse tree
- The experimental results demonstrate that our model is competitive with or outperforms other unsupervised models. In particular, for long reviews, it achieves a competitive or better performance than the supervised models.
- The induced tree shows that the child sentences present additional information about their parent, and the generated summary abstracts for the entire review.

## 2 Proposed Model

In this section, we present our unsupervised end-to-end summarization model with descriptions of *StrSum* and *DiscourseRank*.

### 2.1 StrSum: Structured Summarization

**Model Training:** The outline of *StrSum* is presented in Figure 2.  $y_i$  and  $s_i \in \mathcal{R}^d$  indicate the  $i$ -th sentence and its embedding in a document  $D = \{y_1, \dots, y_n\}$ , respectively.  $w_i^t$  is the  $t$ -th word in a sentence  $y_i = \{w_i^1, \dots, w_i^l\}$ .  $s_i$  is computed via a max-pooling operation across hidden states  $h_i^t \in \mathcal{R}^d$  of the Bi-directional Gated Recurrent Units (Bi-GRU):

$$\vec{h}_i^t = \overrightarrow{\text{GRU}}(\vec{h}_i^{t-1}, w_i^t) \quad (1)$$

$$\overleftarrow{h}_i^t = \overleftarrow{\text{GRU}}(\overleftarrow{h}_i^{t+1}, w_i^t) \quad (2)$$

$$h_i^t = [\vec{h}_i^t, \overleftarrow{h}_i^t] \quad (3)$$

$$\forall m \in \{1, \dots, d\}, s_{i,m} = \max_t h_{i,m}^t \quad (4)$$

Here, we assume that a document  $D$  and its summary compose a *discourse tree*, in which the root is the summary, and all sentences are the nodes. We denote  $a_{ij}$  as the *marginal probability of dependency* where the  $i$ -th sentence is the parent node of the  $j$ -th sentence. In particular,  $a_{0j}$  denotes the probability that a root node is the parent (see Figure 2). We define the probability distribution  $a_{ij}$  ( $i \in \{0, \dots, n\}, j \in \{1, \dots, n\}$ ) as the posterior marginal distributions of a non-projective dependency tree. The calculation of the marginal probability is explained later.

Similar to (Liu and Lapata, 2018), to prevent overload of the sentence embeddings, we decompose them into two parts:

$$[s_i^e, s_i^f] = s_i \quad (5)$$

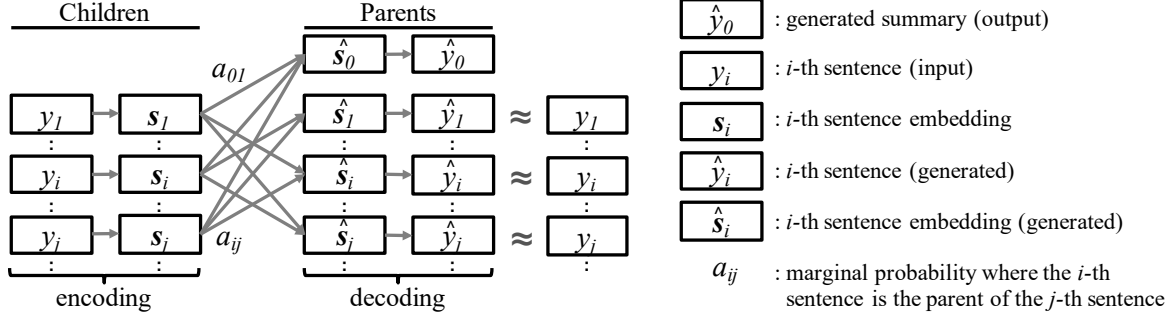


Figure 2: Outline of StrSum.

where the semantic vector  $\mathbf{s}_i^e \in \mathcal{R}^{d_e}$  encodes the semantic information, and the structure vector  $\mathbf{s}_i^f \in \mathcal{R}^{d_f}$  is used to calculate the marginal probability of dependencies.

The embedding of the parent sentence  $\hat{\mathbf{s}}_i$  and that of the *summary*  $\hat{\mathbf{s}}_0$  are defined with parameters  $\mathbf{W}_s \in \mathcal{R}^{d_e * d_e}$  and  $\mathbf{b}_s \in \mathcal{R}^{d_e}$  as:

$$\hat{\mathbf{s}}_i = \tanh \left\{ \mathbf{W}_s \left( \sum_{j=1}^n a_{ij} \mathbf{s}_j^e \right) + \mathbf{b}_s \right\} \quad (6)$$

Using  $\hat{\mathbf{s}}_i$ , the GRU-decoder learns to reconstruct the  $i$ -th sentence, *i.e.*, to obtain the parameters  $\theta$  that maximize the following log likelihood:

$$\sum_{i=1}^n \sum_{t=1}^l \log P(w_i^t | w_i^{<t}, \hat{\mathbf{s}}_i, \theta) \quad (7)$$

**Summary Generation:** An explanation of how the training contributes to the learning of a language model and the gaining of the summary embedding is provided here. As for the former, the decoder learns a language model to generate grammatical sentences by reconstructing the document sentences. Therefore, the model can appropriately decode the summary embedding to  $\hat{y}_0$ .

As for the latter, if the  $j$ -th sentence contributes to generating the  $i$ -th one,  $a_{ij}$  get to be higher. This mechanism models our assumption that *child sentences can generate their parent sentence, but not vice versa*, because the children present additional information about their parent. Hence, the most concise  $k$ -th sentences (*e.g.*, the 1st, 2nd, and 4th in Figure 1), provide less of a contribution to the reconstruction of any other sentences. Thus,  $a_{ik}$  get to be lower for  $\forall i : i \neq 0$ . Because  $a_{ik}$  satisfies the constraint  $\sum_{i=0}^n a_{ik} = 1$ ,  $a_{0k}$  is expected to be larger, and thus the  $k$ -th sentence contributes to the construction of the summary embedding  $\hat{\mathbf{s}}_0$ .

**Marginal Probability of Dependency:** The calculation of the marginal probability of dependency,  $a_{ij}$ , is explained here. We first define the weighted adjacency matrix  $\mathbf{F} = (f_{ij}) \in \mathcal{R}^{(n+1) * (n+1)}$ , where the indices of the first column and row are 0, denoting the root node.  $f_{ij}$  denotes the un-normalized weight of an edge between a parent sentence  $i$  and its child  $j$ . We define it as a pair-wise attention score following (Liu and Lapata, 2018). By assuming a *multi-root* discourse tree,  $f_{ij}$  is defined as:

$$f_{ij} = \begin{cases} \exp(\mathbf{w}_r^\top \mathbf{s}_j^f) & (i = 0 \wedge j \geq 1) \\ \exp(\mathbf{p}_i^\top \mathbf{W}_f \mathbf{c}_j) & (i \geq 1 \wedge j \geq 1 \wedge i \neq j) \\ 0 & (j = 0 \vee i = j) \end{cases} \quad (8)$$

$$\mathbf{p}_i = \tanh(\mathbf{W}_p \mathbf{s}_i^f + \mathbf{b}_p) \quad (9)$$

$$\mathbf{c}_j = \tanh(\mathbf{W}_c \mathbf{s}_j^f + \mathbf{b}_c) \quad (10)$$

where  $\mathbf{W}_f \in \mathcal{R}^{d_f * d_f}$  and  $\mathbf{w}_r \in \mathcal{R}^{d_f}$  are parameters for the transformation.  $\mathbf{W}_p \in \mathcal{R}^{d_f * d_f}$  and  $\mathbf{b}_p \in \mathcal{R}^{d_f}$  are the weight and bias respectively, for constructing the representation of the parent nodes.  $\mathbf{W}_c \in \mathcal{R}^{d_f * d_f}$  and  $\mathbf{b}_c \in \mathcal{R}^{d_f}$  correspond to those of the child nodes.

We normalize  $f_{ij}$  into  $a_{ij}$  based on (Koo et al., 2007).  $a_{ij}$  corresponds to the proportion of the total weight of the spanning trees containing an edge  $(i, j)$ :

$$a_{ij}(\mathbf{F}) = \frac{\sum_{t \in T: (i,j) \in t} v(t|\mathbf{F})}{\sum_{t \in T} v(t|\mathbf{F})} \quad (11)$$

$$= \frac{\partial \log Z(\mathbf{F})}{\partial f_{ij}} \quad (12)$$

$$v(t|\mathbf{F}) = \prod_{(i,j) \in t} f_{ij} \quad (13)$$

$$Z(\mathbf{F}) = \sum_{t \in T} v(t|\mathbf{F}) \quad (14)$$

where  $T$  denotes the set of all spanning trees in a document  $D$ .  $v(t|\mathbf{F})$  is the weight of a tree  $t \in T$ , and  $Z(\mathbf{F})$  denotes the sum of the weights of all trees in  $T$ . From the Matrix-Tree Theorem (Tutte, 1984),  $Z(\mathbf{F})$  can be rephrased as:

$$Z(\mathbf{F}) = |L_0(\mathbf{F})| \quad (15)$$

where  $L(\mathbf{F}) \in \mathcal{R}^{(n+1) \times (n+1)}$  and  $L_0(\mathbf{F}) \in \mathcal{R}^{n \times n}$  are the Laplacian matrix of  $\mathbf{F}$  and its principal submatrix formed by deleting row 0 and column 0, respectively. By solving Eq. 12,  $a_{ij}$  is given by:

$$a_{0j} = f_{0j} [L_0^{-1}(\mathbf{F})]_{jj} \quad (16)$$

$$a_{ij} = f_{ij} [L_0^{-1}(\mathbf{F})]_{jj} - f_{ij} [L_0^{-1}(\mathbf{F})]_{ji} \quad (17)$$

## 2.2 DiscourseRank

StrSum generates the summary under the large influence of the child sentences of the root. Therefore, sentences that are not related to the rating (e.g., the 4th in Figure 1) also affect the summary and can be considered noise. Here, we assume that meaningful sentences (e.g., the 1st and 2nd in Figure 1) typically have more *descendants*, because many sentences provide the explanation of them. Hence, we introduce the *DiscourseRank* to rank the importance of the sentences in terms of the number of descendants. Inspired by PageRank (Page et al., 1999), the DiscourseRank of the root and  $n$  sentences at the  $t$ -th iteration  $\mathbf{r}^t = [r_0, \dots, r_n] \in \mathcal{R}^{(n+1)}$  is defined as:

$$\mathbf{r}^{t+1} = \lambda \hat{\mathbf{A}} \mathbf{r}^t + (1 - \lambda) \mathbf{v} \quad (18)$$

$$\hat{a}_{ij} = \begin{cases} 0 & (i = 0 \wedge j = 0) \\ \frac{1}{n} & (i \geq 1 \wedge j = 0) \\ a_{ij} & (j \geq 1) \end{cases} \quad (19)$$

where  $\hat{\mathbf{A}} = (\hat{a}_{ij}) \in \mathcal{R}^{(n+1) \times (n+1)}$  denotes the stochastic matrix for each dependency,  $\lambda$  is a damping factor, and  $\mathbf{v} \in \mathcal{R}^{(n+1)}$  is a vector with all elements equal to  $1/(n+1)$ . Eq.18 implies that  $r_i$  reflects  $r_j$  more if the  $i$ -th sentence is more likely to be the parent of the  $j$ -th sentence. The  $\mathbf{r}$  solution and updated score of the edge  $(0, j)$   $\bar{a}_{0j}$  ( $j \in \{1, \dots, n\}$ ) are calculated by:

$$\mathbf{r} = (1 - \lambda)(I - \lambda \hat{\mathbf{A}})^{-1} \mathbf{v} \quad (20)$$

$$\bar{a}_{0j} = a_{0j} r_j \quad (21)$$

The updated score  $\bar{a}_{0j}$  is used to calculate the summary embedding  $\hat{s}_0$  instead of Eq.16. As a result, the generated summary reflects the sentences with a higher marginal probability of dependency on the root, while focusing on the main review point.

## 3 Related work

### 3.1 Supervised Review Summary Generation

Several previous studies have addressed abstractive summarization for product reviews (Carenini et al., 2013; Di Fabrizio et al., 2014; Bing et al., 2015; Yu et al., 2016); however, their output summaries are not guaranteed to be grammatical (Wang and Ling, 2016). Neural sequence-to-sequence models have improved the quality of abstractive summarization. Beginning with the adaptation to sentence summarization (Rush et al., 2015; Chopra et al., 2016), several studies have tackled the generation of an abstractive summary of news articles (Nallapati et al., 2016; See et al., 2017; Tan et al., 2017; Paulus et al., 2018). With regard to product reviews, the neural sequence-to-sequence based model (Wang and Ling, 2016) and joint learning with sentiment classification (Ma et al., 2018; Wang and Ren, 2018) have improved the performance of one-sentence summarization. Our work is also based on the neural sequence-to-sequence model, while introducing the new concept of generating the summary by recursively reconstructing a parent sentence from its children.

### 3.2 Unsupervised Summary Generation

Although supervised abstractive summarization has been successfully improved, unsupervised techniques have still not similarly matured. Ganesan et al. (2010) proposed *Opinosis*, a graph-based method for generating review summaries. Their method is word-extractive, rather than abstractive, because the generated summary only contains words that appear in the source document. With the recently increasing number of neural summarization models, Miao and Blunsom (2016) applied a variational auto-encoder for semi-supervised sentence compression. Chu and Liu (2018) proposed *MeanSum*, an unsupervised neural multi-document summarization model for reviews. However, their model is not aimed at generating a summary from a single document and could not directly be extended. Although several previous studies (Fang et al., 2016; Dohare et al., 2018) have used external parsers for unsupervised abstractive summarization, our work, to the best of our knowledge, proposes the first unsupervised abstractive summarization method for a single product review that does not require an external parser.

### 3.3 Discourse Parsing and its Applications

Discourse parsing has been extensively researched and used for various applications. Hirao et al. (2013); Kikuchi et al. (2014); Yoshida et al. (2014) transformed a rhetorical structure theory-based discourse tree (RST-DT; Mann and Thompson, 1988) into a dependency-based discourse tree and regarded the root and the surrounding elementary discourse units as a summary. Gerani et al. (2014) constructed a discourse tree and ranked the aspects of reviews for summarization. Bhatia et al. (2015); Ji and Smith (2017) also constructed a dependency-based discourse tree for document classification. Ji and Smith (2017) pointed out the limitations of using external parsers, demonstrating that the performance depends on the amount of the RST-DT and the domain of the documents.

Against such a background, Liu and Lapata (2018) proposed a model that induces a latent discourse tree without an external corpus. Inspired by structure bias (Cheng and Lapata, 2016; Kim et al., 2017), they introduced *Structured Attention*, which normalizes attention scores as the posterior marginal probabilities of a non-projective discourse tree. The probability distribution of Structured Attention implicitly represents a discourse tree, in which the child sentences present additional information about their parent. We extend it to the unsupervised summarization, *i.e.*, obtaining a summary as the root sentence of a latent discourse tree. While Liu and Lapata (2018) introduce a virtual root sentence and induce a latent discourse tree via supervised document classification, we generate a root sentence via reconstructing a parent sentence from its children without supervision.

## 4 Experiments

In this section, we present our experiments for the evaluation of the summary generation performance of online reviews. The following section provides the details of the experiments and results.<sup>1</sup>

### 4.1 Dataset

Our experiments use the *Amazon product review* dataset (McAuley et al., 2015; He and McAuley, 2016), which contains Amazon online reviews and their one-sentence summaries. It includes 142.8

<sup>1</sup>The code to reproduce the results is available at: <https://github.com/misonuma/strsum>

Domains	Train	Valid	Eval
Toys & Games	27,037	498	512
Sports & Outdoors	37,445	511	466
Movies & TV	408,827	564	512

Table 1: Number of reviews for training (Train), validation (Valid) and evaluation (Eval).

million reviews spanning May 1996 - July 2014. Ma et al. (2018); Wang and Ren (2018) used this dataset for the evaluation of their supervised summary generation model. The same domains considered in their previous work are selected for this study; *Toys & Games*, *Sports & Outdoors*, and *Movies & TV*.

Because our model is trained by identifying and reconstructing a parent sentence from its children, it sometimes fails to construct an appropriate tree for relatively short reviews. It also has a negative influence on summary generation. Therefore, we use reviews with 10 or more sentences for training, and those with 5 or more sentences for validation and evaluation. Table 1 indicates the number of reviews in each domain.

### 4.2 Experimental Details

The source sentences and the summaries share the same vocabularies, which are extracted from the training sources of each domain. We limit a vocabulary to the 50,000 most frequent words appearing in training sets.

The hyper-parameters are tuned based on the performance using the reference summaries in validation sets. We set 300-dimensional word embeddings and initialize them with pre-trained Fast-Text vectors (Joulin et al., 2017). The encoder is a single-layer Bi-GRU with 256-dimensional hidden states for each direction and the decoder is a uni-directional GRU with 256-dimensional hidden states. The damping factor of DiscourseRank is 0.9. We train the model using Ada-grad with a learning rate of  $10^{-1}$ , an initial accumulator value of  $10^{-1}$ , and a batch size of 16. At the evaluation time, a beam search with a beam size of 10 is used.

Similar to (See et al., 2017; Ma et al., 2018), our evaluation metric is the ROUGE-F1 score (Lin, 2004), computed by the *pyrouge* package. We use ROUGE-1, ROUGE-2, and ROUGE-L, which measure the word-overlap, bigram-overlap, and longest common sequence between the reference and generated summaries, respectively.

Domain	Toys & Games			Sports & Outdoors			Movies & TV		
Metric	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
Unsupervised approaches									
TextRank	8.63	1.24	7.26	7.16	0.89	6.39	<b>8.27</b>	<b>1.44</b>	7.35
Opinosis	8.25	1.51	7.52	7.04	1.42	6.45	7.80	1.20	7.11
MeanSum-single	8.12	0.58	7.30	5.42	0.47	4.97	6.96	0.35	6.08
<i>StrSum</i>	11.61	1.56	11.04	9.15	1.38	8.79	7.38	1.03	6.94
<i>StrSum+DiscourseRank</i>	<b>11.87</b>	<b>1.63</b>	<b>11.40</b>	<b>9.62</b>	<b>1.58</b>	<b>9.28</b>	8.15	1.33	<b>7.62</b>
Supervised baselines									
Seq-Seq	13.50	2.10	13.31	10.69	2.02	10.61	7.71	2.18	7.08
Seq-Seq-att	16.28	3.13	16.13	11.49	2.39	11.47	9.05	2.99	8.46

Table 2: ROUGE F1 score of the evaluation set (%). R-1, R-2 and R-L denote ROUGE-1, ROUGE-2, and ROUGE-L, respectively. The best performing model among unsupervised approaches is shown in **boldface**.

### 4.3 Baseline

For the comparisons, two unsupervised baseline models are employed. A graph-based unsupervised sentence extraction method, *TextRank* is employed (Mihalcea and Tarau, 2004), where sentence embeddings are used instead of bag-of-words representations, based on (Rossiello et al., 2017). As an unsupervised word-level extractive approach, we employ *Opinosis* (Ganesan et al., 2010), which detects salient phrases in terms of their redundancy. Because we observe repetitive expressions in the dataset, Opinosis is added as a baseline. Both methods extract or generate a one-sentence summary.

Furthermore, a third, novel unsupervised baseline model *MeanSum-single* is introduced, which is an extended version of the unsupervised neural multi-document summarization model (Chu and Liu, 2018). While it decodes the mean of multiple document embeddings to generate the summary, MeanSum-single generates a single-document summary by decoding the mean of the sentence embeddings in a document. It learns a language model through reconstruction of each sentence. By comparing with MeanSum-single, we verify that our model focuses on the main review points, and does not simply take the average of the entire document.

As supervised baselines, we employ vanilla neural sequence-to-sequence models for abstractive summarization (Hu et al., 2015), following previous studies (Ma et al., 2018; Wang and Ren, 2018). We denote the model as *Seq-Seq* and that with the attention mechanism as *Seq-Seq-att*. The encoder and decoder used are the same as those used in our model.

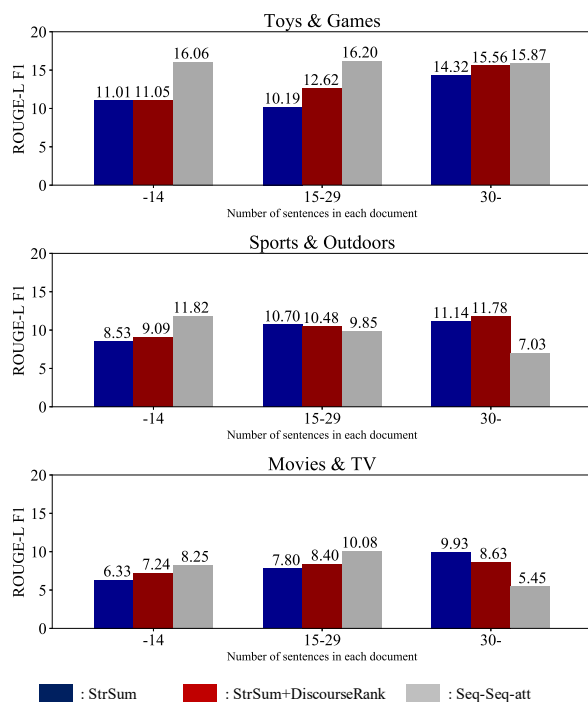


Figure 3: ROUGE-L F1 score on evaluation set with various numbers of sentences.

### 4.4 Evaluation of Summary Generation

Table 2 shows the ROUGE scores of our models and the baselines for the evaluation sets.<sup>2</sup> With regards to *Toys & Games* and *Sports & Outdoors*, our full model (StrSum + DiscourseRank) achieves the best ROUGE-F1 scores among the unsupervised approaches. As for ROUGE-1 and ROUGE-L, two-tailed t-tests demonstrate that the

<sup>2</sup>As Yu et al. (2016); Ma et al. (2018) reported, the reviews and their summaries are usually colloquial and contain more noise than news articles. Therefore, the ROUGE scores on the Amazon review dataset are lower than those obtained for other summarization datasets, such as DUC.

	Generated Summary	Induced Discourse Tree	Sentences in the Main Body
(a)	<ul style="list-style-type: none"> <li>Reference: love this game</li> <li>Seq-Seq-att: fun game</li> <li><b>Our Model (Full):</b> i love this game</li> </ul>	<pre> graph TD     root[root] --- 2[2]     root --- 3[3]     2 --- 1[1]     2 --- 4[4]     3 --- 6[6]     4 --- 5[5] </pre>	<ol style="list-style-type: none"> <li>1. I love this game</li> <li>2. It is so much fun</li> <li>3. I'm all about new and different games</li> <li>4. I love to play this with my brother because he is very bad at keeping score so I win most of the time and he loves to tell each characters story</li> <li>5. And he loves to tell each characters story and to tell why each person got what fate</li> <li>6. It's a must buy if you want a fun and fast card game</li> </ol>
(b)	<ul style="list-style-type: none"> <li>Reference: good value</li> <li>Seq-Seq-att: good for the price</li> <li><b>Our Model (Full):</b> this is a great product for the price</li> </ul>	<pre> graph TD     root[root] --- 1[1]     root --- 6[6]     1 --- 2[2]     1 --- 3[3]     6 --- 4[4]     4 --- 5[5] </pre>	<ol style="list-style-type: none"> <li>1. have not used it yet at the campground but tested it at home and works fine</li> <li>2. use a toothpick to hold the valve open so you can deflate it easily</li> <li>3. if you sit on it and your butt just touches the ground your at the right pressure</li> <li>4. for the price i would recommend it for occasional use</li> <li>5. if your a hard core camper you may want a name brand</li> <li>6. it suits my needs perfectly</li> </ol>
(c)	<ul style="list-style-type: none"> <li>Reference: disappointing</li> <li>Seq-Seq-att: great dvd</li> <li><b>Our Model (Full):</b> this is a great movie</li> </ul>	<pre> graph TD     root[root] --- 1[1]     root --- 4[4]     root --- 7[7]     1 --- 2[2]     4 --- 5[5]     4 --- 3[3]     7 --- 6[6] </pre>	<ol style="list-style-type: none"> <li>1. this had so much potential</li> <li>2. my favorite 3 guitarist yet the sound is muddled</li> <li>3. it should have been recorded in 5</li> <li>4. the video is good</li> <li>5. the sound is horrible though and that 's what makes this a travesty</li> <li>6. i am so disappointed as for concert dvds audio is the most important factor</li> <li>7. not even anamorphic</li> </ol>

Figure 4: Examples of generated summaries and induced latent discourse trees.

difference between our models and the others are statistically significant ( $p < 0.05$ ). Because the abstractive approach generates a concise summary by omitting trivial phrases, it can lead to a better performance than those of the extractive ones. On the other hand, for *Movies & TV*, our model is competitive with other unsupervised extractive approaches; TextRank and Opinosis. One possible explanation is that the summary typically includes named entities, such as the names of characters, actors and directors, which may lead to a better performance of the extractive approaches. For all datasets, our full model outperforms the one using only StrSum. Our models significantly outperform MeanSum-single, indicating that our model focuses on the main review points, and does not simply take the average of the entire document.

Figure 3 shows the ROUGE-L F1 scores of our models on the evaluation sets with various numbers of sentences compared to the supervised baseline model (Seq-Seq-att). For the case of a dataset with less than 30 sentences, the performance of our models is inferior to that of the supervised baseline model. Because our full model generates summaries via learning the latent discourse tree, it sometimes fails to construct a tree, and thus experiences a decline in performance for relatively short reviews. On the other hand, for datasets with the number of sentences exceeding 30, our model achieves competitive or better performance than the supervised model.

## 5 Discussion

### 5.1 Analysis of the Induced Structure

Figure 4 presents the generated summary and the latent discourse tree induced by our full model. We obtained the *maximum spanning tree* from the probability distribution of dependency, using Chu–Liu–Edmonds algorithm (Chu, 1965; Edmonds, 1967).

Figure 4(a) shows the summary and the latent discourse tree for a board game review. Our model generates the summary, "*i love this game*", which is almost identical to the reference. The induced tree shows that the 2nd sentence *elaborates* on the generated summary, while the 3rd sentence provides its *background*. The 4th and 5th sentences *explain* the 1st sentence in detail, *i.e.*, describe why the author loves the game.

Figure 4(b) shows the summary and latent discourse tree of a camping mattress review. Although there is no word-overlap between the reference and generated summary, our model focuses on the positivity in terms of the *price*. On the induced tree, the 1st to 3rd sentences provide a *background* of the summary and mention the high quality of the product. The 6th sentence indicates that reviewer is *satisfied*, while the 4th sentence provides its explanation with regards to the *price*.

In Figure 4(c), we present a failure example of a review of a concert DVD. The reviewer is *disappointed* by the poor quality of the *sound*; however

Toys & Games	<i>StrSum</i>	<i>StrAtt</i>
Projective	38.58%	66.07%
Height	3.06	2.42
Sports & Outdoors	<i>StrSum</i>	<i>StrAtt</i>
Projective	41.26%	58.85%
Height	2.72	2.50
Movies & TV	<i>StrSum</i>	<i>StrAtt</i>
Projective	36.31%	61.20%
Height	3.63	2.37

Table 3: Descriptive statistics for induced latent discourse trees. *StrAtt* denotes the Structured Attention Model (Liu and Lapata, 2018).

our model generates a positive summary, "this is a great movie". The induced tree shows that the sentences describing the high potential (1st), quality of the video (4th), and preference to the picture (7th), all affect the summary generation. Our model regards the *sound* quality as a secondary factor to that of the *video*. Therefore, it fails to prioritize the contrasting aspects; the sound and the video, and generates an inappropriate summary. DiscourseRank cannot work well on this example, because the numbers of sentences mentioning each aspect are not significantly different. To solve such a problem, the aspects of each product must be ranked explicitly, such as in (Gerani et al., 2014; Angelidis and Lapata, 2018).

Table 3 summarizes the characteristics of the induced latent discourse trees. These are compared with those obtained by the Structured Attention model, *StrAtt* (Liu and Lapata, 2018). *StrAtt* induces single-root trees via the document classification task based on the review ratings. For each domain, our model induces more non-projective trees than *StrAtt*. Additionally, the height (the average maximum path length from a root to a leaf node) is larger than that of *StrAtt*. Our model estimates the parent of all the sentences and can induce *deeper* trees in which the edges connect trivial sentences. On the other hand, *StrAtt* identifies salient sentences required for the document classification, and thus induces *shallow* trees that connect the salient sentences and others. As our model prevents the summary from focusing on trivial or redundant sentences by inducing deep and complex trees, it specifically achieves higher performance when considering relatively long reviews.

Reference : 12 yr old son loved it  
*StrSum* : i love it  
*StrSum+DiscourseRank* : my son loves it  
1 : gave this to son for christmas and he immediately started tracing out ...  
2 : he did it every day for two weeks straight  
3 : he would be up in his room or on the living room sofa or at his desk ...  
4 : and he loved coloring different backgrounds  
5 : starry night mountains by day a green forest rushing by  
6 : very fun  
7 : of everything he got for christmas this was probably his favorite gift

(a)

Reference : great goggles , do n't leak .  
*StrSum* : this is a great product for the price  
*StrSum+DiscourseRank* : i have not had any problems with them  
1 : i like the speedo swim goggles  
2 : they have a snug relatively comfortable fit and will not leak ...  
3 : these speedos look good in the water and provide decent visibility  
4 : i find the tinted versions too dark to see underwater  
5 : ... after about 30 uses there are no durability issues so far  
6 : these are nice goggles ... better than the cheap ones ... that leaked all the time

(b)

Figure 5: Visualization of DiscourseRank. The darker the highlighting, the higher the rank score. The references and generated summaries are also shown.

## 5.2 DiscourseRank Analysis

In this section, we demonstrate how DiscourseRank affects the summary generation. Figure 5 visualizes the sentences in the main body and their DiscourseRank scores. We highlight the sentences that achieve a high DiscourseRank score with a darker color.

A review of a car coloring book is presented in Figure 5(a). As expected, the score of the 1st sentence is low, which is not related to the review evaluations, that is, DiscourseRank emphasizes the *evaluative* sentences, such as the 2nd and 6th sentences.

A review of swimming goggles is presented in Figure 5(b). The reviewer is satisfied with the quality of the product. The highlighting shows that DiscourseRank focuses on the sentences that mention leaking (e.g., the 2nd and 5th). While our model (with only *StrSum*) emphasizes the *price* sufficiency, DiscourseRank generates a summary describing that there is no issue with the *quality*.

## 6 Conclusion

In this work, we proposed a novel unsupervised end-to-end model to generate an abstractive summary of a single product review while inducing a latent discourse tree. The experimental results demonstrated that our model is competitive with or outperforms other unsupervised approaches. In



particular, for relatively long reviews, our model achieved competitive or better performance compared to supervised models. The induced tree shows that the child sentences present additional information about their parent, and the generated summary abstracts the entire review.

Our model can also be applied to other applications, such as argument mining, because arguments typically have the same discourse structure as reviews. Our model can not only generate the summary but also identifies the argumentative structures. Unfortunately, we cannot directly compare our induced trees with the output of a discourse parser, which typically splits sentences into elementary discourse units. In future work, we will make comparisons with those of a human-annotated dataset.

## Acknowledgments

We would like to thank anonymous reviewers and members of the Sakata&Mori Laboratory at the Graduate School of Engineering for their valuable feedback. This work was supported by CREST, JST, the New Energy and Industrial Technology Development Organization (NEDO) and Deloitte Tohmatsu Financial Advisory LLC.

## References

- Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3675–3686.
- Parminder Bhatia, Yangfeng Ji, and Jacob Eisenstein. 2015. Better document-level sentiment analysis from rst discourse parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2212–2218.
- Lidong Bing, Piji Li, Yi Liao, Wai Lam, Weiwei Guo, and Rebecca Passonneau. 2015. Abstractive multi-document summarization via phrase selection and merging. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, volume 1, pages 1587–1597.
- Giuseppe Carenini, Jackie Chi Kit Cheung, and Adam Pauls. 2013. Multi-document summarization of evaluative text. *Computational Intelligence*, 29(4):545–576.
- Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 484–494.
- Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.
- Eric Chu and Peter J Liu. 2018. Unsupervised neural multi-document abstractive summarization. *Computing Research Repository*, arXiv:1810.05739v3. Version 3.
- Yoeng-Jin Chu. 1965. On the shortest arborescence of a directed graph. *Scientia Sinica*, 14:1396–1400.
- Giuseppe Di Fabbrizio, Amanda Stent, and Robert Gaizauskas. 2014. A hybrid approach to multi-document summarization of opinions in reviews. In *Proceedings of the 8th International Natural Language Generation Conference*, pages 54–63.
- Shibhansh Dohare, Vivek Gupta, and Harish Karnick. 2018. Unsupervised semantic abstractive summarization. In *Proceedings of ACL 2018, Student Research Workshop*, pages 74–83.
- Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the National Bureau of Standards B*, 71:233–240.
- Yimai Fang, Haoyue Zhu, Ewa Muszyńska, Alexander Kuhnle, and Simone Teufel. 2016. A proposition-based abstractive summariser. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*, pages 567–578.
- Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. 2010. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 340–348.
- Shima Gerani, Yashar Mehdad, Giuseppe Carenini, Raymond T Ng, and Bitia Nejat. 2014. Abstractive summarization of product reviews using discourse structure. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1602–1613.
- Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, pages 507–517.
- Tsutomu Hirao, Yasuhisa Yoshida, Masaaki Nishino, Norihito Yasuda, and Masaaki Nagata. 2013. Single-document summarization as a tree knapsack problem. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1515–1520.

- Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. Lcsts: A large scale chinese short text summarization dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1967–1972.
- Masaru Isonuma, Toru Fujino, Junichiro Mori, Yutaka Matsuo, and Ichiro Sakata. 2017. Extractive summarization using multi-task learning with document classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2101–2110.
- Yangfeng Ji and Noah A Smith. 2017. Neural discourse structure for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 996–1005.
- Armand Joulin, Edouard Grave, and Piotr Bojanowski Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2, pages 427–431.
- Yuta Kikuchi, Tsutomu Hirao, Hiroya Takamura, Manabu Okumura, and Masaaki Nagata. 2014. Single document summarization based on nested tree structure. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 315–320.
- Yoon Kim, Carl Denton, Luong Hoang, and Alexander M Rush. 2017. Structured attention networks. In *Proceedings of the 5th International Conference on Learning Representations*.
- Terry Koo, Amir Globerson, Xavier Carreras, and Michael Collins. 2007. Structured prediction models via the matrix-tree theorem. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 141–150.
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, volume 8.
- Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In *Mining text data*, pages 415–463. Springer.
- Yang Liu and Mirella Lapata. 2018. Learning structured text representations. *Transactions of the Association of Computational Linguistics*, 6:63–75.
- Shuming Ma, Xu Sun, Junyang Lin, and Xuancheng Ren. 2018. A hierarchical end-to-end model for jointly improving text summarization and sentiment classification. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4251–4257.
- William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.
- Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52.
- Yishu Miao and Phil Blunsom. 2016. Language as a latent variable: Discrete generative models for sentence compression. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 319–328.
- Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411.
- Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.
- Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations*.
- Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.
- Gaetano Rossiello, Pierpaolo Basile, and Giovanni Semeraro. 2017. Centroid-based text summarization through compositionality of word embeddings. In *Proceedings of the MultiLing Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, pages 12–21.
- Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.
- Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1073–1083.

- Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1171–1181.
- William Thomas Tutte. 1984. *Graph theory*, volume 21. Addison-Wesley.
- Hongli Wang and Jiangtao Ren. 2018. A self-attentive hierarchical model for jointly improving text summarization and sentiment classification. In *Proceedings of the 10th Asian Conference on Machine Learning*, pages 630–645.
- Lu Wang and Wang Ling. 2016. Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57.
- Yasuhisa Yoshida, Jun Suzuki, Tsutomu Hirao, and Masaaki Nagata. 2014. Dependency-based discourse parser for single-document summarization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1834–1839.
- Naitong Yu, Minlie Huang, Yuanyuan Shi, and Zhu Xiaoyan. 2016. Product review summarization by exploiting phrase properties. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 1113–1124.