Information Systems 94 (2020) 101615

Contents lists available at ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/is

Knowledge-guided unsupervised rhetorical parsing for text summarization

Shengluan Hou [a,b,*], Ruqian Lu [a,c]

[a] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[b] University of Chinese Academy of Sciences, Beijing 100049, China
[c] Academy of Mathematics and Systems Sciences & Key Lab of MADIS, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:
Received 5 July 2019
Received in revised form 10 July 2020
Accepted 28 July 2020
Available online 3 August 2020
Recommended by Carsten Binnig

Keywords:
Automatic text summarization
Rhetorical structure theory
Domain knowledge base
Attentional encoder–decoder
Natural language processing

ABSTRACT

Automatic text summarization (ATS) has recently achieved impressive performance thanks to recent advances in deep learning and the availability of large-scale corpora. However, there is still no guarantee that the generated summaries are grammatical, concise, and convey all salient information as the original documents have. To make the summarization results more faithful, this paper presents an unsupervised approach that combines rhetorical structure theory, deep neural model, and domain knowledge concern for ATS. This architecture mainly contains three components: domain knowledge base construction based on representation learning, the attentional encoder–decoder model for rhetorical parsing, and subroutine-based model for text summarization. Domain knowledge can be effectively used for unsupervised rhetorical parsing thus rhetorical structure trees for each document can be derived. In the unsupervised rhetorical parsing module, the idea of translation was adopted to alleviate the problem of data scarcity. The subroutine-based summarization model purely depends on the derived rhetorical structure trees and can generate content-balanced results. To evaluate the summary results without golden standard, we proposed an unsupervised evaluation metric, whose hyper-parameters were tuned by supervised learning. Experimental results show that, on a large-scale Chinese dataset, our proposed approach can obtain comparable performances compared with existing methods.

1. Introduction

Automatic text summarization (ATS) aims to produce a condensed representation while keeping the salient elements from one or a group of topic-related documents, which is a potential research area receiving considerable attentions from academia to industry. With the amounts of data are being generated in the Web age, ATS plays an increasingly important role in addressing the problem of how to acquire information and knowledge in a fast, reliable, and efficient way. Generally, ATS can be categorized into two types: extractive summarization and abstractive counterpart [1,2]. Extractive text summarization approaches directly extract salient textual units to produce the summary. Conversely, abstractive models paraphrase the salient contents using natural language generation (NLG) techniques.

Abstractive methods concern the generation of new sentences, new phrases while retaining the same meaning as the same source documents have. We have seen the recent progress in abstractive ATS [3,4]. However, there is still no guarantee that the generated summaries are grammatical and convey absolutely the same meaning as the original documents have. On the other hand, extractive approaches are typically more intuitive yet they often generate verbose contents with unnecessarily long sentences and redundant information. To make the results more faithful and coherent, motivated by the discourse structure theories [5,6], we incorporate discourse structures into extractive summarization generation. In this way, the discourse structures can be leveraged to make the results less verbose while retaining extractive.

Discourse structure theories involve understanding the part-whole nature of textual documents. The task of rhetorical parsing, for example, involves understanding how two text spans are related to each other in the context. As Web mining extracts latent knowledge from the Web content [7], rhetorical parsing reveals the meaningful knowledge out of vast amounts of documents and help in improving many NLP applications. The theoretical foundation is the rhetorical structure theory (RST) [6], which a comprehensive theory of discourse organization. RST investigates how clauses, sentences and even larger text spans connect together as a whole. RST assumes that discourse is not merely a collection of random utterances but the discourse units connect

* Corresponding author at: Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China.
E-mail address: houshengluan1989@163.com (S. Hou).

https://doi.org/10.1016/j.is.2020.101615
0306-4379/

to each other as a whole in a logical and topological way. RST explains text coherence by postulating a hierarchical, connected tree-structure (denote as RS-tree) for a given text [8], in which every part has a role, a function to play, with respect to other parts in the text. RST has been empirically proved useful for improving the performance of NLP tasks that need to combine meanings of larger text units, such as single-document summarization [9,10], QA and chabot [11] and text classification [5,12], etc.

In this paper, we focus on the extractive single-document summarization task, the key to which is how to score the salience of candidate text summary units (i.e. sentences, clauses, etc.). Our proposed model can benefit from RST for more faithful results, which mainly consists of three components: domain knowledge base construction based on representation learning, the attentional encoder–decoder model for rhetorical parsing, and subroutine-based model for text summarization. The first component extracts domain keywords based on representation learning. The domain keywords contain three types: acting agents, major influence factors, and dynamics of a domain. The second component leverages the output of the first component for RS-tree construction, which will be fed to the third component for the summary generation.

Our aim is Chinese-oriented text summarization. To alleviate the problem of data scarcity, we leverage the labeled English data RST-DT [13] and map texts of English and Chinese into the same latent space, from which the rhetorical relation between two Chinese text spans can be determined. Furthermore, the last component extracts summary texts from the derived RS-trees. The generated summary from our subroutine-based text summarization model can be always balanced between the nucleus and satellite subtrees.[1]

The contributions of this work can be concluded as follows:

- We first proposed an unsupervised Chinese-oriented rhetorical parsing method. Existing rhetorical parsing methods are English-oriented, supervised methods that often trained on RST-DT, a human-annotated discourse treebank of WSJ articles under the framework of RST. Our proposed method leverages the idea of translation and embeds the Chinese and English texts in the same latent space. In this way, the rhetorical relations between Chinese text spans can be determined by the rhetorical relations in RST-DT.
- Domain knowledge was utilized in the rhetorical parsing procedure, which was constructed based on representation learning. Domain knowledge was used in two aspects: one for discourse segmentation and the other one for guiding rhetorical structure inference. Furthermore, the attention mechanism was adopted in rhetorical parsing thus the attention weights enable our model has the ability to focus on relevant and drown-out irrelevant parts of the input.
- Different from the majority of literature, our subroutine-based summarization model is purely based on the generated rhetorical structure. The basic processing unit is the elementary discourse unit (EDU), which is relatively shorter than sentence. Thus the generated summary can be more informative. This model is based on 'importance first' principle, each time the 'currently' most important EDU from the rhetorical structure will be selected one by one mechanically. The 'importance first' principle makes the selection of EDUs alternated between nucleus and satellite subtrees. Thus the generated summary can always be balanced.
- We also proposed an unsupervised summarization evaluation metric. This evaluation metric considers many aspects of how faithful a generated summary is. To make this evaluation metric more effective, the hyper-parameters were tuned by supervised learning on the golden standard of DUC2002.

The remainder of this paper is organized as follows. Section 2 reviews some related works, including approaches about domain knowledge, rhetorical parsing and automatic text summarization. Section 3 is the domain knowledge base construction based on representation learning. The large-scale Chinese dataset and experimental results on it will also be given. The unsupervised rhetorical parsing approach is elaborated in Section 4, in which the idea of translation and attention mechanism was adopted. Section 5 is about the subroutine-based text summarization. An unsupervised summarization evaluation metric and experimental results are shown in Section 6. The paper is concluded with a summary and an outlook for further research in Section 7.

2. Related works

In this section, we briefly review some related works. In Section 2.1, we will first discuss the works about domain knowledge. Section 2.2 then introduces rhetorical structure theory, which is an important theoretical foundation of our work. Finally, the latest and classical approaches of automatic text summarization will be described in Section 2.3.

2.1. Domain knowledge

Knowledge is power. Domain knowledge plays a significant role in many NLP tasks. For instance, the knowledge graph (KG) is a knowledge base proposed by Google to enhance its search engine's results with information gathered from a variety of sources. Li and Mao [14] proposed an effective way of combining human knowledge and information from data for CNN to achieve better performance. They presented K-CNN: a knowledge-oriented CNN for causal relation extraction. In K-CNN, the convolutional filters are automatically generated based on WordNet and FrameNet. The data-oriented channel is used to learn other important features of causal relation from the data. Lu et al. [15] studied the concepts of big knowledge, big-knowledge system, and big-knowledge engineering. Ten massiveness characteristics for big knowledge and big-knowledge systems are defined and explored. Zheng [16] explored how to enable humans to use big knowledge correctly and effectively in the biomedical domain. There are also some knowledge-based text summarization methods, we refer to [17,18].

Domain knowledge keyword extraction is defined as the task that automatically identifies a set of the terms that best describe the domain of documents [19]. Generally, domain keyword extraction approaches can be divided into two categories as unsupervised methods and supervised methods. TF–IDF is one of the simplest unsupervised approaches. The top-k high TF–IDF value words are chosen as keywords. Until now, TF–IDF remains a strong unsupervised baseline [20]. TextRank [21] is another typical unsupervised method, which formulates keyword extraction as "recommendation". The supervised methods often take keyword extraction as classification problems [22]. However, a number of annotated dataset is needed. These methods remain limited when only unlabeled data is available. Kong et al. [23] constructed a Chinese sentiment lexicon using representation learning. A skip-gram model was built to predict word embeddings according to the context words and their composing characters, whose outputs were then fed into a Random Forest (RF)

[1] In RST, nucleus and satellite play different roles to the writer's purpose. In general, what nucleus of a rhetorical relation expresses is more essential than what satellite expresses; The nucleus is comprehensible independent of the satellite, but not vice versa.

classifier. Words of the same polarity were then grouped together to form the sentiment lexicon.

With regard to KG, YAGO is automatically extracted from Wikipedia and other sources. YAGO2 [24] contains 447 million facts about 9.8 million entities, in which an article in Wikipedia becomes an entity. DBpedia [25] extracts fact triples from 111 different language versions of Wikipedia. To tackle the problem of low recall for pattern-based approaches, Angeli et al. [26] leveraged dependency parsing tree for relation triple extraction. They constructed a few patterns for canonically structured sentences, and shift the focus to a classifier that learns to extract self-contained clauses from long sentences. On the other hand, the key idea of KG embedding is to embed components of a KG into continuous vector spaces and thus to simplify the manipulation while preserving the inherent structure of the KG [27]. Typical methods contain TransE [28], TransH [29], TransR [30], etc. KG embedding has been applied to and benefits a wide variety of downstream NLP tasks such as KG completion, question answering, and so on.

2.2. Rhetorical structure theory

Rhetorical structure theory [6] is a comprehensive theory of text organization. With more and more attentions on this theory, RST has been applied to many high-level NLP applications since Marcu's earlier works on RST parsing and applications on text summarization [31]. RST is now one of the most popular theories for discourse analysis.

Central to RST is rhetorical relation, which exists between two neighboring text units. The interpretation of how text spans are semantically related to each other described by rhetorical relations is crucial to retrieve important information from documents. There are two types of rhetorical relations: mononuclear relations and multi-nuclear relations. In the former ones, one of the text spans is more important than the other one, which play the role of nucleus and satellite respectively. One the other hand, all text spans are equally salient in multi-nuclear relations, which all play the role of nucleus.

According to RST, the minimum processing unit is EDU. EDU acts as a syntactic constituent that has independent semantics. In this sense, an EDU corresponds to a clause or a simple sentence. RST explains text coherence by postulating a hierarchical, connected tree-structure (i.e. RS-tree) for a given text. In the RS-tree, each leaf node corresponds to an EDU. Each internal node corresponds to a larger text span which captures the rhetorical relation between its two children.

Rhetorical parsing aims to generate EDU sequences and RS-trees for given documents. It involves finding roles for every granularity of text spans and rhetorical relations that hold between them. There are rule-based methods, traditional machine learning methods, and deep learning methods. LeThanh et al. [32] used syntactic information and cue phrases to segment sentences and integrated constraints about textual adjacency and textual organization to generate best RS-trees. Tofiloski et al. [33] presented syntactic rules and a lexical rule-based discourse segmenter (SLSeg). Soricut and Marcu's SPADE model [34] used two probabilistic models for sentence-level analysis, one for segmentation and the other for RS-tree building. After that, most research focused on SVM-based discourse analysis. They regarded relation identification as classification problem [35,36]. Joty et al. [37] first used Dynamic Conditional Random Field (DCRF) for sentence-level discourse analysis, and then proposed a two-stage rhetorical parser. Recent advances in deep learning led to further progress in rhetorical parsing. DPLP [38] is a representation learning method, whose main idea is to project lexical features into a latent space. DPLP constructs RS-trees in a shift-reduce way. A multi-class linear SVM classifier was learned to decide whether shift or reduce operation would be taken. Li et al.'s recursive method [39] contains two components. The first is to obtain the distributed representation for sentences using recursive convolution based on its syntactic tree. The second component contains two classifiers, one is used for determining whether two adjacent nodes should be merged. If so, the other one selects the appropriate rhetorical relation to the newly merged subtree.

2.3. Automatic text summarization

Automatic text summarization has spurred a surge of research and experimentation since its remarkable effect in the modern Web age. With the fast development of deep learning technologies, many efforts applied encoder–decoder models into ATS. The usage of attention mechanism into text summarization was first brought to prominence by Rush et al. [40]. This attentional encoder–decoder abstractive model was trained on large-scale Gigaword[2] dataset. Its variants and further improvements include [41,42] et al. Neural extractive methods are also popular, such as pointer network-based models [43,44], SummaRuNNer [42], SWAP-NET [45], HIBERT [46], JECS [47], etc. Most of the extractive models are trained on CNN/DM[3] dataset. However, large-scale dataset is necessary for these neural models since they are purely data-driven. Note that PacSum [48] is a typical unsupervised, directed text graph-based extractive model, in which BERT [49] was employed as sentence encoder to compute sentence similarity for better measuring sentence centrality.

Besides the above deep learning-based approaches, there are also some other solutions, such as lexical chain-based approaches [50], classical machine learning-based approaches [51], graph-based unsupervised methods [21], optimization-based methods [52], etc. The ILP-based method did exact inference under a maximum coverage model [52]. The classical traditional machine learning-based methods take TF–IDF, n-gram, the position, and others as features to extract summary sentences. For more details, we refer to [1,2]. Graph-based methods have become increasingly prevalent and far-reaching since their easy implementation and relatively good performance, such as Textrank [21]. Note that this type of approach remains unsupervised. Another representative of the unsupervised algorithm is SummCoder [53], whose summary sentence selection module contains three metrics: sentence content relevance is measured by a deep auto-encoder network, sentence novelty is measured by sentence similarity based on sentence embeddings and sentence position relevance is derived by a hand-designed score function.

The authors of RST have long speculated that the nuclei in RS-tree constitute an adequate summarization of the text. It was first validated by Marcu [54]. Louis et al. [10] proved that the structure features (i.e. position in the global structure of the whole text) of RS-tree are the most useful feature to compute the salience of text spans. Hirao et al. [9] treated the summary generation as a tree knapsack problem. They transformed an RS-tree into a dependency-based discourse tree (DEP-DT), which can be directly used to take tree trimming approaches for text summarization. For MDS, to address the redundancy problem, Zahri et al. [55] used RS-trees for cluster-based MDS. They utilized rhetorical relations that exist between two sentences to group similar sentences into multiple clusters to identify themes of common information, from which candidate summary sentences were extracted. In this paper, we propose a further contribution to this approach, focusing on unsupervised extractive summarization.

[2] https://catalog.ldc.upenn.edu/ldc2003t05.

[3] https://github.com/deepmind/rc-data.

3. Domain knowledge base construction based on representation learning

Domain knowledge plays a significant role in many NLP tasks. At present, most of the existing knowledge bases are in the form of knowledge graph, such as YAGO, DBpedia, etc, which generally consists of entity and relation triples. The knowledge triples in a KG are composed of two entities along with their relation, each of which is in the form of $\langle e1, r1, e2\rangle$, where $e1$ and $e2$ are entities that often nouns or noun phrases, $r1$ is the relation between $e1$ and $e2$. However, knowledge keywords for a domain are also indispensable. For a domain, knowledge keywords can provide a panorama for this domain. In this section, we propose a framework of constructing the domain knowledge base on the basis of representation learning. Our proposed domain knowledge contains three types of keywords: acting agents, major influence factors, and dynamics of a domain.

We define the domain as:

Definition 1 (*Domain*)**.** A domain is a particular area of human knowledge. Such as education, finance, et al.

For a domain, keywords can be regarded as the knowledge generalization of the full text in corresponding literature and help readers to quickly grasp the core idea, core technique, or core methodology, etc. In general, two different domains have different knowledge keywords, but maybe with some common knowledge keywords. The definition of domain knowledge keyword is given in Definition 2.

Definition 2 (*Domain Knowledge Keyword, DKK*)**.** A domain knowledge keyword is a basic and characteristic element of this domain, which is represented by a word or phrase and is often referred to when talking about some aspects of this domain. DKK can generalize the main topics of domain texts.

Example 1. "Teacher", "Student", "Professor", "Teach", "Learn", "Library", "Course", "Doctoral" are DKKs of the domain "Education".

Two different domains may share some of their DKKs (e.g. "Library" may be a DKK of some other domains), but never share their whole sets of DKKs. The less the size of the shared DKKs, the more are the two domains different from each other.

We argue that besides nouns, verbs, and adjectives (adverbs) also serve as the key components. In the above example, "Teacher" is a noun, "Teach" is a verb, and "Doctoral" is an adjective. In fact, these three types of keywords constitute the main types of DKKs. For each domain, we construct domain knowledge from large-scale texts. The DKB in this work is composed of a set of triples containing domain keywords.

Definition 3 (*Domain Knowledge Base, DKB*)**.** For a domain, the DKB can be represented by a triple:

$$\langle A, P, T\rangle \tag{1}$$

where

- A denotes nouns and named entities, each of which represents the acting agents of this domain;
- P acts as the major influence factors of this domain, which are nouns;
- T denotes the concepts about the dynamics of this domain, each of which is often adjective or adverb.

These three types of keywords constitute a full DKB for a domain.

Our goal is to construct the DKB for each domain in a fast and efficient manner. Traditional methods can obtain high accuracy but with low recall. Moreover, many efforts are needed when using them in a new domain. On the other hand, word embedding becomes more and more popular since its robustness and efficiency. It can be trained on large-scale dataset without any other extra resource. With the availability of large-scale corpora, word embedding has demonstrated its powerfulness in many NLP applications [56]. However, the primitive methods, such as Word2Vec [57,58], has the following drawbacks:

- They are completely data-driven, which implies that the performance is sensitive to the training set and the dimension of vector to some extent;
- Although such results are semantically informative, they disregard the valuable information (e.g. semantic relations) contained in semantic lexicons such as HowNet [59] and WordNet [60];
- The learned embeddings may not be suited for the task of interest.

To further improve the performance, recently, much further research progress towards incorporating human-knowledge into the training of word embedding and has indicated that word embedding can benefit from human-knowledge. In this work, we leverage the representation learning from human-knowledge-enriched word embedding methods for DKB construction.

Definition 4 (*Domain Knowledge Base Construction, DKBC*)**.** Given a large set of documents that consists of texts for several domains $\{D_1, D_2, \ldots, D_t\}$, DKBC aims to extract a DKB from each domain texts $D_i (1 \leq i \leq t)$, the constructed DKB is in the form of (1).

Specially, for each domain D_i, given the corresponding documents $\{d_1, d_2, \ldots, d_k\}$, DKBC can automatically generate three types of DKKs as defined in Definition 2. All generated DKKs can constitute the DKB (denote as DKB_i) in the form of (1) such that:

- If $w_m \in DKB_i$, then $w_m \in DICT_i$, where $DICT_i$ is the vocabulary taken from D_i;
- Suppose $DKB_i = \langle A_i, P_i, T_i\rangle$, $w_p \in A_i$, $w_q \in P_i$, then $p \neq q$.

To obtain better results, our model is an integration of three different models. The first one is the representation learning-based model, which we call VWRank. The other two models are the TF–IDF model and the TextRank model. TF–IDF is an important indicator of the word's saliency. TextRank is an "recommendation" strategy for voting salient words.

3.1. The architecture of VWRank

Our DKBC model utilizes representation learning from word embedding approaches. We use the improved word representation learning with sememes method, called SE-WRL [61]. The sememe knowledge base they used is Hownet [59]. SE-WRL provides different strategies, among which SE-WRL-SAT achieved the best performance according to their original paper. SE-WRL-SAT learns not only the original word embeddings for context words, but sememe embeddings for target words.

For each domain, we use SE-WRL-SAT to learn word representations. Then we define the similarity between two candidate words as cosine distance.

Motivated by TextRank, the score of candidate keyword cw_i can be computed as:

$$Score(cw_i) = (1-d) + d * \sum_{cw_j \in S(cw_i)} \times \frac{Sim(cw_j, cw_i)}{\sum_{cw_k \in S(cw_j)} Sim(cw_j, cw_k)} Score(cw_j) \tag{2}$$

Table 1
Excerpts of url and its corresponding domain in SouCA dataset.

Url	Domain
http://www.xinhuanet.com/world/	国际 (World)
http://news.china.com/zh_cn/international/	国际 (World)
http://finance.sina.com.cn/	财经 (Finance)
http://sports.china.com/	体育 (Sports)
http://china.soufun.com/	房产 (House)
...	...

where $d \in (0, 1)$ is a damping factor, which has the role of integrating into the model the probability of jumping from a given candidate word to another random candidate word. $S(cw_i)$ and $S(cw_j)$ are two sets of candidate words that cw_i and cw_j similar with, respectively. After several iterations, the $Score(cw_i)$ can converge to a fixed value.

3.2. Model integration

Besides VWRank, in a domain, we also calculate the TF–IDF and TextRank values for candidate words in each domain. We denote the high score candidate keywords of VWRank, TF–IDF and TextRank as C_{vw}, C_{ti} and C_{tr}. The final score of a candidate keyword is computed as:

$$Score(cw_i) = \alpha * I(C_{wv}, cw_i) + \beta * I(C_{ti}, cw_i) + \gamma * I(C_{tr}, cw_i) \quad (3)$$

where α, β and γ are harmonic coefficients, $I(\cdot)$ is the indicator function such that

$$I(C, cw) = \begin{cases} 1, & \text{if } cw \in C \\ 0, & \text{else} \end{cases} \quad (4)$$

Then the final DKKs are composed of candidate keywords that further filtered by the value of:

$$p_{tt}(cw_i) = \frac{\text{The number of documents that } cw_i \text{ presents}}{\text{The number of all documents in this domain}} \quad (5)$$

Finally, all selected DKKs will be organized in hierarchies by their semantic in Hownet.

3.3. Dataset: SogouCA

SogouCA[4] is a large-scale Chinese corpus, which is crawled and provided by Sogou Labs from dozens of Chinese news websites, including news reports and reviews.

Each document in SogouCA contains fields of "url", "docno", "contenttitle", and "content". Leveraging "url" information, we can categorize documents into corresponding domains. Excerpts of "url" and its corresponding domain are shown in Table 1.

After that, we collected texts for 15 domains. We did preprocessing including delete empty or very short lines, ignore extremely long lines, etc. The statistics of documents in each domain are listed in Table 2, from which we can see that most of the domains contain tens of thousands of documents.

Unlike English, to manipulate text at the word level, word segmentation is needed for Chinese text processing. We used HanLP [62] for Chinese word segmentation, part-of-speech tagging, and named entity recognition (NER), which is a Chinese natural language processing tool.

4 https://www.sogou.com/labs/resource/ca.php.

Fig. 1. The DKKs of "Finance" domain.

Fig. 2. The DKKs of "IT" domain.

3.4. Experimental results of DKBC

We have finished the DKBC for 15 domains according to the above methods. The derived 15 DKBs can be used for other NLP applications [63]. The statistics of DKB in each domain are shown in Table 2.

Fig. 1 shows some DKKs of "Finance" domain in Chinese. The DKKs contains keywords such as "中国 (China)"(Agent), "投资者 (investor)"(Phenomenon), "风险 (risk)" (Phenomenon), "上涨 (increase)"(Tendency), etc. These words can provide a panorama for domain "Finance".

On the other hand, in the domain of "IT" (Fig. 2), some DKKs are "微软 (Microsoft)"(Agent), "排行 (Ranking)"(Phenomenon), "市场 (market)"(Phenomenon), "上市 (Be listed)"(Tendency), etc. These two examples can validate the effectiveness of our method.

4. Unsupervised rhetorical parsing

Rhetorical structure theory was proposed as a way to attribute structure to text, which often represents a text as a tree structure. It is characterized by rhetorical relations, which reflect the semantic and functional judgments about the text spans they connect. We first give a formal definition of the rhetorical structure tree.

Definition 5 (*Rhetorical Structure Tree*)**.** Rhetorical Structure Tree (RS-tree) is a tree representation of a document under the framework of RST. The leaf nodes of an RS-tree are EDUs. Each internal node is characterized by a rhetorical relation and corresponds to a contiguous text span. The siblings are connected via a rhetorical relation such that in most cases one is nucleus and the other is satellite. The siblings are both nuclei when they are connected by a multi-nuclear relation.

In Definition 5, EDU is the minimal textual unit of an RS-tree, which means that it cannot be split into smaller text spans. EDU acts as a syntactic constituent that has independent semantics. In this sense, an EDU functionally corresponds to a simple sentence or a clause in a complex sentence.

Table 2
The statistics of documents and DKB of each domain in the SogouCA dataset. "# X" means the number of X. For instance, "Avg. #Words." means the average number of words in a document, "#Agents" means the number of agent keywords in a domain.

Domain	Sports	IT	Military	Olympic	Culture	House	Domestic	Entertainment
#Docs	323,861	22,033	17,607	74,374	7212	22,381	2454	93,949
Avg. #Sens.	16.49	20.58	19.27	17.09	23.49	17.73	19.46	16.63
Avg. #Words.	334.65	493.37	447.82	385.83	505.17	229.97	490.63	372.77
#Agents	2687	622	561	1706	179	308	126	1318
#Phenomenons	6162	1580	1186	3416	439	996	267	4422
#Tendencies	5290	1004	767	2316	230	564	141	3448
Domain	Auto	Finance	Lady	Health	Education	Society	World	Total
#Docs	44,462	263,575	72,970	5712	53,197	2698	2566	1,009,231
Avg. #Sens.	18.48	22.21	15.49	20.04	21.22	20.60	14.25	18.55
Avg. #Words.	413.14	492.01	277.30	332.22	418.76	419.98	333.31	391.75
#Agents	533	2856	637	56	902	93	130	12,723
#Phenomenons	1999	5469	3504	413	2298	278	204	32,633
#Tendencies	1440	3308	3011	247	1579	135	139	23,619

Definition 6 (*Rhetorical Parsing*)**.** Rhetorical Parsing, also called RST analysis, RST parsing, or rhetorical analysis, is a procedure of generating EDU sequences and deriving RS-trees for given texts. It involves segmenting discourse into EDUs and finding roles for every granularity of text spans (EDUs, sentences, paragraphs, and even larger spans) and rhetorical relations that hold between them.

As depicted in Definition 6, rhetorical parsing contains two steps: discourse segmentation and RS-tree construction. An example RS-tree for a given Chinese text is shown in Fig. 3. The leaf nodes numbered with digits are four EDUs. The internal nodes correspond to text spans are characterized by rhetorical relations (such as Joint and Elaboration). The arrow from A to B denotes A and B are satellite and nucleus respectively in the sense of that relation. They are both nuclei when multi-nuclear relation exists between A and B. Horizontal lines correspond to text spans, and vertical lines identify text spans which are nuclei.

4.1. Discourse segmentation based on domain knowledge base

Leveraging domain knowledge, we segment each document in each domain into an EDU sequence. According to Definition 5, EDU functionally corresponds to a simple sentence or a clause. We first segment a text into paragraphs and further sentences by punctuations. Then DKB is used for segmenting sentences into EDUs. Concretely, for a domain, given its DKB K_d and domain texts T_d, for each text $t_i^d \in T_d$, Algorithm 1 is the detailed segmentation algorithm.

Algorithm 1 Discourse Segmentation for a Domain Text

Input: A document t_i^d; DKB K_d.
Output: EDU sequences S_{edu}.
1: Segment t_i^d into sentence sequence S by punctuations (line break for segmenting into paragraphs, period, question mark, etc for segmenting into sentences).
2: **for** each sentence s_j in S **do**
3: Scan s_j and match their words against the domain keywords in K_d;
4: If the domain keywords of a clause have the form "A+P+T" or "P+T", then put it into S_{edu}.
5: **end for**
6: Output the EDUs in S_{edu} according to their order in the original text.

After Algorithm 1, each derived EDU is a part of a sentence or clause, characterizing the domain relatedness of its elements. Moreover, most EDUs have the form of "A+P+T" with respect to domain keywords. For the form of "P+T", we borrow an agent keyword from the nearest neighbor EDU to form a complete triple. After that, each EDU has a DKB triple $\langle a, p, t\rangle$.

4.2. Rhetorical structure theory discourse treebank

For RS-tree construction, existing models contain classical machine learning-based methods and deep learning-based methods, almost all of which are supervised methods. These approaches were trained on Rhetorical Structure Theory Discourse Treebank (RST-DT) [13]. RST-DT was developed as a human-annotated discourse-level corpus with RS-trees for 385 English-written Wall Street Journal texts. These texts were manually annotated by the professional language analysts grounded in the framework of RST. There are 78 fine-grained rhetorical relations that grouped into 18 coarse-grained relation categories. In the existing approaches, the latter 18 categories are often used for training and testing. Since there exist multi-nuclear relations, non-binary relations are often converted into a cascade of right-branching binary relations for convenience. In RST-DT, there are 21,789 EDUs and 21,404 text pairs that are characterized by rhetorical relations.

4.3. Attentional encoder–decoder model for RS-tree construction

The objective of RS-tree construction is to find rhetorical relations between two adjacent text spans(including EDUs). Then the RS-tree can be constructed in a bottom-up way. For our Chinese-oriented rhetorical parsing work, there is no human-annotated Chinese-oriented discourse treebank like RST-DT in English. Motivated by the basic ideas of recent progress in unsupervised machine translation [64], we propose to leverage RST-DT and embed Chinese text spans and English text spans into the same latent space. Thus the rhetorical relation between two Chinese text spans can be derived by the rhetorical relations in RST-DT. Our work is unsupervised since there is no labeled Chinese dataset is used. The architecture of rhetorical relation identification is shown in Fig. 4.

The unsupervised rhetorical parsing model we propose is composed of two encoders, a decoder, and two classifiers. The translation encoder is responsible for encoding Chinese and English texts into a latent space and the DKB encoder is used for representing domain keyword sequence. The attention-based decoder 1 and attention-based decoder 2 are with the same parameters, whose only difference is the choice of lookup tables when applying them to different languages. The two classifiers are used for rhetorical relation identification.

In Fig. 4, the components in dotted box are to constrain the model can map text pair from Chinese (English) to English (Chinese). Suppose $\langle L1T1, L1T2\rangle$ is English (Chinese) text pair,

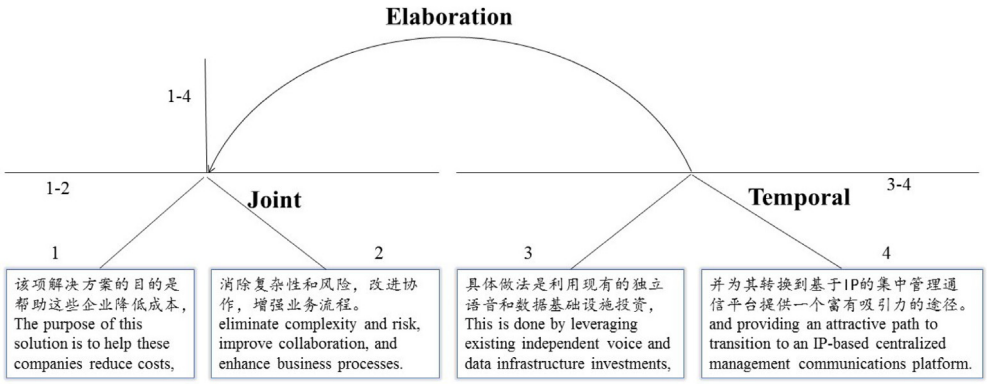

Fig. 3. An example of RS-tree.

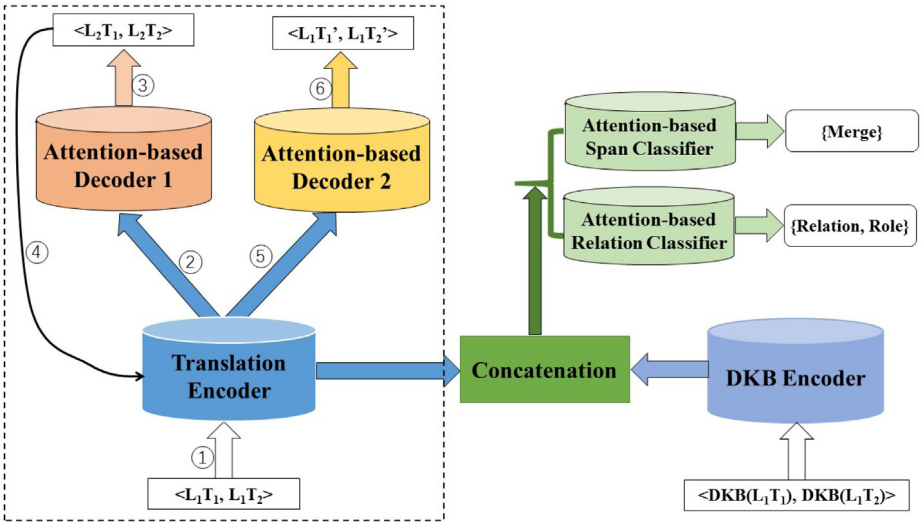

Fig. 4. The architecture of rhetorical relation identification.

the output of attention-based decoder 1 is Chinese (English) $\langle L2T1, L2T2\rangle$, which then will be input to the translation encoder. The output of attention-based decoder 2 is English (Chinese) $\langle L1T1', L1T2'\rangle$. The object of this procedure is to learn a mapping such that translations are close in the same latent space. The translation loss function is:

$$L_{trans} = \sum[\Delta(\langle L1T1, L1T2\rangle, \langle L1T1', L1T2'\rangle) + \Delta(\langle L2T1, L2T2\rangle, \langle L2T1', L2T2'\rangle)] \quad (6)$$

where Δ is the sum of token-level cross-entropy losses.

The second objective of our model is to train two classifiers. When $\langle L1T1, L1T2\rangle$ is English text pair, we denote $\langle DKB(L1T1), DKB(L1T2)\rangle$ as its domain keyword sequence. The concatenation of the two encoders' hidden states is fed into two classifiers. The attention-based span classifier is used for determining whether $L1T1$ and $L1T2$ should be merged into a new subtree and if so the attention-based relation classifier is used to assign which relation and which role should be labeled to the merged node and its two children respectively. The loss function used for classification is also cross-entropy loss.

In this work, our proposed model is based on the sequence-to-sequence model with attention [65]. These two encoders are both bidirectional-GRU which returns a sequence of hidden states whereas the decoder is also a GRU, which takes as input the previous hidden state, the current word and a context vector given by a weighted sum over the encoder states.

The final loss function is:

$$L = \lambda_{trans} L_{trans} + \lambda_{clas} L_{clas} \quad (7)$$

where λ_{trans} and λ_{clas} are hyper-parameters, L_{clas} is classification loss.

For inference, the input is Chinese EDU text pair along with their domain keywords and the output is (1): whether they can be merged into a subtree; (2) If so, which rhetorical relation and which role should be labeled to the merged node. Then the text of the merged node and its neighboring node's text will form the new input text pair. Loop this step until an RS-tree for a document has been constructed.

5. Subroutine-based model for automatic text summarization

In this section, we present a subroutine-based model for automatic text summarization, which has been introduced in our previous paper [66]. Different from the majority of literature, our subroutine-based summarization model is purely based on the generated RS-tree from Section 4.

The basic processing unit is EDU, which is relatively shorter than sentence. Thus the generated summary can be more informative than the summary that composed of sentences. The summarization algorithm is based on 'importance first' principle, each time the 'currently' most important EDU from RS-tree will be selected one by one mechanically. In this way, we can obtain a hierarchy of different summarizations level-wise from simple to

complex by adding one more EDU at each level. There two ways to control the complexity of the summarized results: either by specifying the word length limit or the rate of text reduction.

When going to produce a summary, the summarization model traverses the RS-tree in a nucleus preference way. That is: (1) a nucleus node is always preferred over its sibling satellite node; (2) if node A is preferred over node B, then all child nodes of A are preferred over B; (3) the selection of EDUs should be alternated between the left and right subtrees of the root node whenever both subtrees are not empty. Whenever a leaf node (EDU) is traversed, the text unit represented by it will be put to the final summary. In our model, a nucleus node is always preferred over its sibling satellite node. It is balanced in the sense that nodes of the two subtrees of the root node are scanned alternately until one of the two subtrees is completely traversed.

All details of this subroutine-based text summarization algorithm are clarified in Algorithm 2 (Main framework), Algorithm 3 (Subroutine dfinding), Algorithm 4 (Subroutine index), Algorithm 5 (Subroutine ufinding), and Algorithm 6 (Subroutine sfinding). The $nuc(x)$ and $sat(x)$ mean the nucleus resp. satellite child nodes of x. $flip(x)$ is the flip-flop function with $flip(0) = 1$ and $flip(1) = 0$. *dfp* and *sfp* are pointers pointing to the entry of Subroutine *dfinding* resp. Subroutine *sfinding*. The *zp*. is a formal parameter for storing a pointer. For example, after the subroutine call $dfinding(0, R(T), sat(R(T)), dfp)$ it is $zp = dfp$ in the subroutine body of *dfinding*.

Algorithm 2 Subroutine-based Text Summarization

Input: The RS-tree $R(T)$ of a document T; Summary length/cadence ratio r; $j := 1$; $k := 1$.
Output: The generated EDU sequences R_{edu}.

```
if R(T) is a leaf node then
    Put R(T) into R_edu; Stop, success.
else
    Call dfinding(0, R(T), sat(R(T)), dfp).
end if
```

The subroutine *index* (Algorithm 4) controls the algorithm can terminate when meeting the halting condition.

After the success of the algorithm, the final summary can be derived by sorting the EDUs in R_{edu} according to their order in the original text.

6. Experimental results

6.1. Training details about unsupervised rhetorical parsing

The training of our unsupervised rhetorical parsing was carried out on SogouCA and RST-DT datasets. We used a mini-batch stochastic gradient descent (SGD) algorithm together with Adam [67] with an initial learning rate of 0.001 to train this model. In each epoch, the training data in each batch are the mixture of Chinese and English text pairs. We used Textrank for the DKBC of RST-DT. The size of word embedding for both language and GRU hidden state dimensions are set to 100 and 300 respectively. For two decoders, texts are generated using greedy decoding.

6.2. Unsupervised quantitative evaluation metric

The commonly used evaluation metric for text summarization is ROUGE [68]. ROUGE evaluates n-gram co-occurrences between summary pairs. It works by comparing an automatically produced summary against a set of reference summaries. The reference summaries are typically produced by human linguists. It is a very expensive and time-consuming process. It is even more difficult

Algorithm 3 Subroutine $dfinding(w, x, y, zp)$

```
if w = 0 then
    if j = 0 then Stop, success.
    else
        if nuc(x) is a non-leaf node then
            Call dfinding(w, nuc(x), y, zp).
        else
            Put nuc(x) into R_edu; Call index; Call zp.(flip(w), nuc(x), y, sfp).
        end if
    end if
else
    if k = 0 then Stop, success.
    else
        if y is a leaf node then
            Put y into R_edu; k := 0.
            if j = 0 then stop, success.
            else
                Call index; Call zp.(flip(w), x, y, sfp).
            end if
        else
            if nuc(y) is non-leaf node then
                Call dfinding(w, x, nuc(y), zp).
            else
                Put nuc(y) into R_edu; Call index; Call zp.(flip(w), x, nuc(y), sfp).
            end if
        end if
    end if
end if
```

Algorithm 4 Subroutine *index*

```
if the total length of EDUs in R_edu satisfies r then
    Stop, success.
end if
```

Algorithm 5 Subroutine $ufinding(w, x, y, zp)$

```
if w = 0 then
    if j = 0 then stop, success.
    else
        Call zp.(flip(w), parent(x), y, sfp).
    end if
else
    if k = 0 then stop, success.
    else
        Call zp.(flip(w), x, parent(y), sfp).
    end if
end if
```

Table 3
Evaluation results of our proposed system for automatic text summarization by comparing it to other baselines on SogouCA dataset using our proposed evaluation metric. Summary length limit of 50 words, 100 words and summary ratio of 10%, 20% are reported.

Approaches	10%	20%	50 words	100 words
Lead	62.9	72.8	63.2	73.6
TextRank [21]	65.2	76.6	66.1	75.4
ILP [52]	66.7	79.5	66.9	79.7
SummCoder [53]	68.8	82.3	69.0	82.5
PacSum [48]	69.8	81.9	70.4	82.7
Ours	**71.1**	**85.7**	**72.6**	**86.3**

Algorithm 6 Subroutine *sfinding*(w, x, y, zp)

if $w = 0$ **then**
  if $j = 0$ **then**
    Call *sfinding*(*flip*(w), x, y, *sfp*).
  else if *parent*$(x) = R(T)$ **then**
    $j := 0$.
    if $k = 0$ **then** then stop, success.
    else
      Call *sfinding*(*flip*(w), x, y, *sfp*).
    end if
  else if *sibling*(x) has been travelled **then**
    Call *ufinding*(w, x, y, sfp).
  else
    if *sibling*(x) is a leaf node **then**
      Put *sibling*(x) into R_{edu}; Call *index*; Call *ufinding*(w, x, y, sfp).
    else
      Call *dfinding*$(w$, *sibling*(x), y, *sfp*$)$.
    end if
  end if
else
  if $k = 0$ **then**
    Call *sfinding*(*flip*(w), x, y, *sfp*).
  else if *parent*$(y) = R(T)$ **then**
    $k := 0$.
    if $j = 0$ **then** stop, success.
    else
      Call *sfinding*(*flip*(w), x, y, *sfp*).
    end if
  else if *sibling*(y) has been travelled **then**
    Call *ufinding*(w, x, y, sfp)
  else
    if *sibling*(y) is a leaf node **then**
      Put *sibling*(y) into R_{edu}; Call *index*; Call *ufinding*(w, x, y, sfp).
    else
      Call *dfinding*$(w, x$, *sibling*(y), *sfp*$)$.
    end if
  end if
end if

when facing large amounts of texts in the big data age. There is no reference summary as the golden standard in our selected dataset (i.e. SogouCA).

Based on the assumption that good summaries will tend to be similar to the input in terms of content, Louis and Nenkova [69,70] measured the distribution of terms between the source text and its summary. To better measure the semantic overlap between source documents and the generated summaries, Gao et al. [71] propose to use BERT [49] to develop unsupervised evaluation method, i.e. SUPERT. The salient information was first identified as the pseudo reference summary. The semantic overlap between the pseudo reference summary and the generated summary can be measured. To build a quantization standard, we propose an unsupervised evaluation metric based on the previous approaches. We consider that a faithful summary should:

1. Overlaps with title in three aspects: n-gram, domain knowledge keywords, and named entities; These metrics can be used to measure how much of the generated summary encapsulates the original texts.
2. Contains more domain knowledge keywords than other non-summary texts; As mentioned in Section 3, domain knowledge keywords can provide a panorama for a domain. We evaluate this in two aspects: the domain knowledge keywords that both in summary and title, the domain knowledge keywords that appear in the generated summary.
3. Contains more named entities than other non-summary texts; We obtain this by: (1) computing the number of unique NEs that appear in the original text divided by the number of unique NEs that appear in the summary; (2) computing the number of unique NEs that appear in the original text divided by the number of unique NEs that appear both in the summary and title.
4. The similarities between two summary EDU texts should be lower in case of redundancy. We measure the redundancy by using the average ROUGE score between each two of the summary EDUs.

Formally, for a document d in domain D, whose title is t, the faithful score of a generated summary s is computed as:

$$Score(s) = \mathbf{w} \cdot [ROUGE(t, s), \frac{Count_{dkb}(t, s)}{Count_{dkb}(d)}, \frac{Count_{ent}(t, s)}{Count_{ent}(d)}, \frac{\sum_{e_1, e_2 \in s} ROGUE(e_1, e_2)}{Count_{edu}(s)}, \frac{Count_{dkb}(s)}{Count_{dkb}(d)}, \frac{Count_{ent}(s)}{Count_{ent}(d)}] + b \tag{8}$$

where $\mathbf{w} \in \mathbb{R}^N$, b is a scalar. $ROUGE(a, b)$ denotes the ROUGE score between text a and b. $Count_{dkb}(a, b)$ ($Count_{ent}(a, b)$) denotes the number of domain keywords (named entities) that a and b both have. $Count_{dkb}(x)$ ($Count_{ent}(x)$) denotes the number of domain keywords (named entities) that x has. $Count_{edu}(s)$ denotes the number of EDUs in s. To make the score more objective, the hyper-parameters [$\mathbf{w}$, b] were learned using linear regression on DUC2002[5] dataset. DUC2002 is an official evaluation dataset for automatic text summarization, which is high-quality since all of the reference summaries were written by human linguists. In the training step, the faithful score for each golden standard is set to 100.

6.3. Results and analysis

For each generated RS-tree, we applied Algorithm 2 for summary generation. In what follows, we present the results using our method and our comparison to previous works. Since our model is unsupervised, we compare it with existing unsupervised single-document summarization methods. The baselines include:

- **Lead** selects the leading sentences in the document until length limit to form a summary, which is often used as an official baseline of DUC. Previous works have implied that the Lead method provides a strong baseline.
- **TextRank** [21] is a graph-based text summarization model. It represents the document as a graph in which sentences are nodes and the edges between two sentences are connected based on the similarity between them.
- **ILP** [52] is a text summarization technique which utilizes Integer Linear Program (ILP) for inference under a maximum coverage model.
- **SummCoder** [53] is an unsupervised framework for extracting sentences based on deep auto-encoders.
- **PacSum** [48] leverages a directed graph for measuring sentence centrality, in which the sentence similarity was computed using the sentence representation derived from BERT [49].

[5] https://www-nlpir.nist.gov/projects/duc/data/2002_data.html.

Table 4
Case study with summary ratio = 20%.

Title 空客称争取以合作方式参与中国大飞机项目 Airbus said it is seeking to participate in China's large aircraft project in a cooperative manner.
Lead 新华网天津5月30日电 空中客车中国公司总裁博龙在天津接受新华社记者独家采访时说，空客正与中方合作伙伴商议，争取以合作方式参与中国大飞机项目。对于中国正在研发的大飞机项目，空客正与中方合作伙伴商议争取以合作方式参与该项目。 Xinhuanet Tianjin, May 30th – Bolong, the president of Airbus China, said in an exclusive interview with Xinhua News Agency in Tianjin that Airbus is negotiating with Chinese partners to participate in China's large aircraft project in a cooperative manner. For China's developing large aircraft project, Airbus is negotiating with Chinese partners to participate in the project in a cooperative manner.
TextRank 空中客车中国有限公司企业资讯部提供的情况:早在1999年，空中客车公司与中国航空工业第一集团公司签署协议，计划分阶段向中国转让A320系列飞机机翼制造技术和生产线，目标是到2007年底使中国能够为空中客车在英国布劳顿和北威尔士的工厂制造A320系列飞机完整的机翼结构。 Airbus China Ltd. Corporate Information Department provided: As early as 1999, Airbus and China Aviation Industry First Group signed an agreement to transfer the A320 series aircraft wing manufacturing technology and production line to China in stages. The goal is to enable China to manufacture the complete wing structure of the A320 family of aircraft for Airbus' plants in Broughton and North Wales, England at the end of 2007.
ILP 对于中国正在研发的大飞机项目，空客正与中方合作伙伴商议争取以合作方式参与该项目。中国作为世界航空市场增长最快的国家，已成为空中客车和波音全球的竞争焦点。通过该中心，中国已承担空中客车于2005年10月6日正式发起的、最新的A350飞机项目5%的工作份额。 For China's large aircraft project, Airbus is negotiating with Chinese partners to participate in the project in a cooperative manner. As the fastest growing country in the world aviation market, China has become the competition focus of Airbus and Boeing. Through the center, China has undertaken the 5% share of the latest A350 aircraft project, which was officially launched by Airbus on October 6, 2005.
SummCoder 对于中国正在研发的大飞机项目，空客正与中方合作伙伴商议争取以合作方式参与该项目。双方均不断加大在华采购、投资和技术合作，双方在最新机型上的重要零部件生产，也都有中国参与。博龙认为中国的大飞机之路困难而漫长，“空客花了近40年才取得今天的成就，现在我们拥有实力雄厚的工业基地。” For China's large aircraft project, Airbus is negotiating with Chinese partners to participate in the project in a cooperative manner. Both parties have continuously increased procurement, investment and technical cooperation in China. Both parties have also participated in the production of important parts and components on the latest models. Bolong holds that the road of China's large aircraft is difficult and long. "Airbus has spent nearly 40 years to achieve today's achievements, and now we have a powerful industrial base".
PacSum 新华网天津5月30日电 空中客车中国公司总裁博龙在天津接受新华社记者独家采访时说，空客正与中方合作伙伴商议，争取以合作方式参与中国大飞机项目。中国作为世界航空市场增长最快的国家，已成为空中客车和波音全球的竞争焦点。 Xinhuanet Tianjin, May 30th – Bolong, the president of Airbus China, said in an exclusive interview with Xinhua News Agency in Tianjin that Airbus is negotiating with Chinese partners to participate in China's large aircraft project in a cooperative manner. As the fastest growing country in the world aviation market, China has become the competition focus of Airbus and Boeing.
Ours 空客正与中方合作伙伴商议争取以合作方式参与中国大飞机项目。中国作为世界航空市场增长最快的国家，已成为空中客车和波音全球的竞争焦点。双方均不断加大在华采购、投资和技术合作，双方在最新机型上的重要零部件生产，也都有中国参与。 Airbus is negotiating with Chinese partners to participate in China's large aircraft project in a cooperative manner. As the fastest growing country in the world aviation market, China has become the competition focus of Airbus and Boeing. Both parties have continuously increased procurement, investment and technical cooperation in China. Both parties have also participated in the production of important parts and components on the latest models.

We generated four versions of summary (word length limit = 100, 200 and the rate of text summarization = 10%, 20%). Table 3 shows the faithful score of our method and baseline approaches. Our proposed framework outperforms many of the existing text summarizers on SogouCA dataset in terms of our proposed faithful score such as ILP, graph-based approaches. The reason lies in: (1) We take EDU as the basic processing unit, which is relatively shorter than sentence. Thus the generated summary can be more informative. (2) The RS-tree guides the process of summary generation. (3) Our proposed subroutine-based summary extraction model is based on the 'importance first' principle. The most salient textual units can be identified as the summary texts.

The final summaries obtained from a sample SogouCA document by each summarizer (i.e. Lead, TextRank, ILP, SummCoder, and Ours) with summary ratio = 20% are shown in Table 4. From the summaries, it can be observed that the result generated from our method is more informative than other methods. The result from our proposed model is a summarization of the results that derived from other models. The summary generated by ILP is similar to that generated by SummCoder but it is different from those generated by TextRank. Regarding the deep learning-based models, PacSum performs slightly better than SummCoder. Our proposed model obtains the best results than others.

7. Concluding remarks

In this paper, we proposed a novel unsupervised rhetorical parsing architecture for single-document extractive summarization. The proposed approach mainly contains three parts: domain knowledge base construction, Chinese-oriented rhetorical parsing and level-wise extractive summarization. To the best of our knowledge, this is the first study to adopt translation idea for rhetorical parsing.

Firstly, we proposed a domain knowledge base construction model based on representation learning. The learned DKB can provide a panorama for a domain, which has two important roles for rhetorical parsing. One is discourse segmentation, and the other one is guiding rhetorical relation identification. In the unsupervised rhetorical parsing model, we leveraged the idea of translation and designed a novel attention-based sequence-to-sequence model for rhetorical relation identification. Then the subroutine-based ATS model can accept different word length limit or summarization ratio and provide content-balanced results based on RS-tree. To evaluate our generated summary results in an unsupervised way, we presented a faithful score, whose hyper-parameters were learned on the DUC2002 dataset.

Directions for future work are many and varied. One of the challenges left for the future is to further improve the performance of rhetorical parsing. Such as introducing attribute grammar into the deep neural model. Another important further work would be to utilize RS-tree for multi-document summarization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank the developers of Pytorch [72]. This work was supported by the National Key Research and Development Program of China under grant 2016YFB1000902; and the National Natural Science Foundation of China (No. 61232015, 61472412, and 61621003).

References

[1] C.C. Aggarwal, Text summarization, in: Machine Learning for Text, Springer, 2018, pp. 361–380.
[2] M. Gambhir, V. Gupta, Recent automatic text summarization techniques: a survey, Artif. Intell. Rev. 47 (1) (2017) 1–66.
[3] S. Chopra, M. Auli, A.M. Rush, 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, 2016, pp. 93–98.
[4] W.-T. Hsu, C.-K. Lin, M.-Y. Lee, K. Min, J. Tang, M. 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), 2018, pp. 132–141.
[5] Y. Ji, N.A. Smith, Neural discourse structure for text categorization, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 996–1005.
[6] W.C. Mann, S.A. Thompson, Rhetorical structure theory: Toward a functional theory of text organization, Text-Interdiscip. J. Study Discourse 8 (3) (1988) 243–281.
[7] I. Yaqoob, I.A.T. Hashem, A. Gani, S. Mokhtar, E. Ahmed, N.B. Anuar, A.V. Vasilakos, Big data: From beginning to future, Int. J. Inf. Manage. 36 (6) (2016) 1231–1247.
[8] D. Das, Signalling of Coherence Relations in Discourse (Ph.D. thesis), Simon Fraser University, 2014.
[9] T. Hirao, Y. Yoshida, M. Nishino, N. Yasuda, M. Nagata, Single-document summarization as a tree knapsack problem, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1515–1520.
[10] A. Louis, A. Joshi, A. Nenkova, Discourse indicators for content selection in summarization, in: Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Association for Computational Linguistics, 2010, pp. 147–156.
[11] B. Galitsky, D. Ilvovsky, Chatbot with a discourse structure-driven dialogue management, in: Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics, 2017, pp. 87–90.
[12] M. Kraus, S. Feuerriegel, Sentiment analysis based on rhetorical structure theory: Learning deep neural networks from discourse trees, Expert Syst. Appl. 118 (2019) 65–79.
[13] L. Carlson, D. Marcu, M.E. Okurowski, Building a discourse-tagged corpus in the framework of rhetorical structure theory, in: Current and New Directions in Discourse and Dialogue, Springer, 2003, pp. 85–112.
[14] P. Li, K. Mao, Knowledge-oriented convolutional neural network for causal relation extraction from natural language texts, Expert Syst. Appl. 115 (2019) 512–523.
[15] R. Lu, X. Jin, S. Zhang, M. Qiu, X. Wu, A study on big knowledge and its engineering issues, IEEE Trans. Knowl. Data Eng. (2018).
[16] L. Zheng, Applications of Big Knowledge Summarization (Ph.D. thesis), New Jersey Institute of Technology, 2018.
[17] A. Goldstein, Y. Shahar, An automated knowledge-based textual summarization system for longitudinal, multivariate clinical data, J. Biomed. Inform. 61 (2016) 159–175.
[18] A. Timofeyev, B. Choi, Building a knowledge based summarization system for text data mining, in: International Cross-Domain Conference for Machine Learning and Knowledge Extraction, Springer, 2018, pp. 118–133.
[19] A. Onan, S. Korukoğlu, H. Bulut, Ensemble of keyword extraction methods and classifiers in text classification, Expert Syst. Appl. 57 (2016) 232–247.
[20] L. Marujo, W. Ling, I. Trancoso, C. Dyer, A.W. Black, A. Gershman, D.M. de Matos, J. Neto, J. Carbonell, Automatic keyword extraction on twitter, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2015, pp. 637–643.
[21] R. Mihalcea, P. Tarau, Textrank: Bringing order into text, in: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, 2004.
[22] S.K. Bharti, K.S. Babu, A. Pradhan, Automatic keyword extraction for text summarization in multi-document e-newspapers articles, Eur. J. Adv. Eng. Technol. 4 (6) (2017) 410–427.
[23] L. Kong, C. Li, J. Ge, Y. Yang, F. Zhang, B. Luo, Construction of microblog-specific chinese sentiment lexicon based on representation learning, in: Pacific Rim International Conference on Artificial Intelligence, Springer, 2018, pp. 204–216.
[24] J. Hoffart, F.M. Suchanek, K. Berberich, G. Weikum, Yago2: A spatially and temporally enhanced knowledge base from wikipedia, Artificial Intelligence 194 (2013) 28–61.
[25] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P.N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer, et al., Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia, Semant. Web 6 (2) (2015) 167–195.
[26] G. Angeli, M.J.J. Premkumar, C.D. Manning, Leveraging linguistic structure for open domain information extraction, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, pp. 344–354.
[27] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, IEEE Trans. Knowl. Data Eng. 29 (12) (2017) 2724–2743.
[28] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: Advances in Neural Information Processing Systems, 2013, pp. 2787–2795.
[29] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014.
[30] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.
[31] D. Marcu, The Theory and Practice of Discourse Parsing and Summarization, MIT press, 2000.
[32] H. Lethanh, G. Abeysinghe, C. Huyck, Generating discourse structures for written texts, in: International Conference on Computational Linguistics, 2004, p. 329.
[33] M. Tofiloski, J. Brooke, M. Taboada, A syntactic and lexical-based discourse segmenter, in: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, Association for Computational Linguistics, 2009, pp. 77–80.
[34] R. Soricut, D. Marcu, Sentence level discourse parsing using syntactic and lexical information, in: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, 2003, pp. 149–156.
[35] V.W. Feng, G. Hirst, Text-level discourse parsing with rich linguistic features, in: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, Association for Computational Linguistics, 2012, pp. 60–68.
[36] H. Hernault, H. Prendinger, M. Ishizuka, et al., Hilda: A discourse parser using support vector machine classification, Dialogue Discourse 1 (3) (2010).
[37] S. Joty, G. Carenini, R. Ng, Y. Mehdad, Combining intra-and multi-sentential rhetorical parsing for document-level discourse analysis, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2013, pp. 486–496.
[38] Y. Ji, J. Eisenstein, Representation learning for text-level discourse parsing, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2014, pp. 13–24.
[39] J. Li, R. Li, E. Hovy, Recursive deep models for discourse parsing, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 2061–2069.
[40] A.M. Rush, S. Chopra, J. Weston, A Neural Attention Model for Abstractive Sentence Summarization, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 379–389.
[41] J. Lin, X. Sun, S. Ma, Q. Su, Global encoding for abstractive summarization, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2018, pp. 163–169.
[42] R. Nallapati, F. Zhai, B. Zhou, Summarunner: A recurrent neural network based sequence model for extractive summarization of documents, in: Thirty-First AAAI Conference on Artificial Intelligence, 2017.
[43] J. Cheng, M. Lapata, Neural summarization by extracting sentences and words, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 484–494.
[44] A. See, P.J. Liu, C.D. Manning, Get to the point: Summarization with pointer-generator networks, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 1073–1083.

[45] A. Jadhav, V. Rajan, Extractive summarization with SWAP-NET: Sentences and words from alternating pointer networks, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018, pp. 142–151.
[46] X. Zhang, F. Wei, M. Zhou, Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization, 2019, arXiv preprint arXiv:1905.06566.
[47] J. Xu, G. Durrett, Neural extractive text summarization with syntactic compression, 2019, arXiv preprint arXiv:1902.00863.
[48] H. Zheng, M. Lapata, Sentence centrality revisited for unsupervised summarization, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 6236–6247.
[49] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.
[50] S. Hou, Y. Huang, C. Fei, S. Zhang, R. Lu, Holographic lexical chain and its application in chinese text summarization, in: Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data, Springer, 2017, pp. 266–281.
[51] R. Ferreira, L. de Souza Cabral, R.D. Lins, G.P. e Silva, F. Freitas, G.D. Cavalcanti, R. Lima, S.J. Simske, L. Favaro, Assessing sentence scoring techniques for extractive text summarization, Expert Syst. Appl. 40 (14) (2013) 5755–5764.
[52] D. Gillick, B. Favre, A scalable global model for summarization, in: Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing, 2009, pp. 10–18.
[53] A. Joshi, E. Fidalgo, E. Alegre, L. Fernández-Robles, Summcoder: An unsupervised framework for extractive text summarization based on deep auto-encoders, Expert Syst. Appl. (2019).
[54] D. Marcu, From discourse structures to text summaries, Intell. Scalable Text Summ. (1997).
[55] N.A.H. Zahri, F. Fukumoto, M. Suguru, O.B. Lynn, Exploiting rhetorical relations to multiple documents text summarization, Int. J. Netw. Secur. Appl. 7 (2) (2015) 1.
[56] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, Learning word vectors for 157 languages, in: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), 2018.
[57] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, 2013, arXiv preprint arXiv:1301.3781.
[58] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems, 2013, pp. 3111–3119.
[59] D. Zhendong, D. Qiang, Hownet and the Computation of Meaning (with Cd-rom), World Scientific, 2006.
[60] C. Fellbaum, Wordnet, in: Theory and Applications of Ontology: Computer Applications, Springer, 2010, pp. 231–243.
[61] Y. Niu, R. Xie, Z. Liu, M. Sun, Improved word representation learning with sememes, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 2049–2058.
[62] H. He, HanLP: Han language processing, 2014, URL https://github.com/hankcs/HanLP.
[63] R. Lu, S. Hou, On semi-supervised multiple representation behavior learning, 2019, arXiv preprint arXiv:1910.09292.
[64] G. Lample, A. Conneau, L. Denoyer, M. Ranzato, Unsupervised machine translation using monolingual corpora only, in: International Conference on Learning Representations (ICLR), 2018.
[65] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: International Conference on Learning Representations 2015, 2015.
[66] R. Lu, S. Hou, C. Wang, Y. Huang, C. Fei, S. Zhang, Attributed rhetorical structure grammar for domain text summarization, 2019, arXiv preprint arXiv:1909.00923.
[67] S. Ruder, An overview of gradient descent optimization algorithms, 2016, arXiv preprint arXiv:1609.04747.
[68] C.-Y. Lin, Rouge: A package for automatic evaluation of summaries, Text Summ. Branches Out (2004).
[69] A. Louis, A. Nenkova, Automatically evaluating content selection in summarization without human models, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2009, pp. 306–314.
[70] A. Louis, A. Nenkova, Automatically assessing machine summary content without a gold standard, Comput. Linguist. 39 (2) (2013) 267–300.
[71] Y. Gao, W. Zhao, S. Eger, SUPERT: towards new frontiers in unsupervised evaluation metrics for multi-document summarization, 2020, arXiv preprint arXiv:2005.03724.
[72] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, 2017.