

Knowledge-aware Document Summarization: A Survey of Knowledge, Embedding Methods and Architectures

Yutong Qu^{a,*}, Wei Emma Zhang^{a,*}, Jian Yang^b, Lingfei Wu^c, Jia Wu^b

^a*School of Computer Science, The University of Adelaide, SA 5005, Australia*

^b*School of Computing, Macquarie University, NSW 2109, Australia*

^c*JD.COM Silicon Valley Research Center, CA 94043, USA*

Abstract

Knowledge-aware methods have boosted a range of natural language processing applications over the last decades. With the gathered momentum, knowledge recently has been pumped into enormous attention in document summarization, one of natural language processing applications. Previous works reported that knowledge-embedded document summarizers excel at generating superior digests, especially in terms of informativeness, coherence, and fact consistency. This paper pursues to present the first systematic survey for the state-of-the-art methodologies that embed knowledge into document summarizers. Particularly, we propose novel taxonomies to recapitulate knowledge and knowledge embeddings under the document summarization view. We further explore how embeddings are generated in embedding learning architectures of document summarization models, especially of deep learning models. At last, we discuss the challenges of this topic and future directions.

Keywords: Knowledge, Knowledge embedding, Document summarization

2010 MSC: 00-01, 99-00

*Corresponding author

Email addresses: yutong.qu@adelaide.edu.au (Yutong Qu),
wei.e.zhang@adelaide.edu.au (Wei Emma Zhang)

1. Introduction

With the exponential burst of textual data, demands in condensing voluminous text contents have been ubiquitous, bringing document summarization one of the most immensely researched fields in Natural Language Processing (NLP). Document Summarization (DS) aims to generate an abridged version of single or multiple topic-related texts as concise and coherent as possible while preserving the salient and factually consistent information [1]. The document summarization task with a single input document is known as the Single Document Summarization (SDS). By contrast, the Multi-Document Summarization (MDS) task emphasizes synthesizing a large number of topic-related documents to generate a compressed summary from various times and perspectives. In addition, there are two general methods in document summarization: 1) the Extractive Document Summarization (EDS) method respects the lexicon of the original text, regarding the summary formation is verbatim by key words and phrases selected from the source corpus; and 2) the Abstractive Document Summarization (ADS) method respects the semantics of the original text, regarding the summary construction is by rephrasing texts according to the comprehension of text substances. Generally, a document summarization model is to achieve the following goals [2]:

- **G1.** Coverage: A document summarization model aims to generate a comprehensive summary that covers all the main and noteworthy contents of the input text(s);
- **G2.** Non-redundancy: A document summarization model aims to generate a precise and concise summary without any redundant or meaninglessly repeated information;
- **G3.** Readability: A document summarization model aims to generate a smooth and logical summary composed by human-readable and coherent sentences to the viewer;

For multi-document summarization models, an additional goal is [3]:

- **G4.** Relevancy: A multi-document summarization model aims to identify related information within multiple input texts while generating the summary.

Recently, knowledge utilization in the summarization models has exhibited a huge potential for promoting the summarizer performance in terms of **G1** to **G4** and fueled one more document summarization capacity target:

- **G5.** Factual Consistency: A document summarization model aims to generate a consistent summary that obeys text facts and the commonsense of the real world.

The goal **G5** also reflects that the knowledge refers to the information acquired from facts and commonsense in source corpora and external sources, which normally can be captured in knowledge graphs [4]. In contrast to novel auxiliary knowledge, such as the timeline or visual information [5], the knowledge, in general, focuses more on human knowledge from a linguistic perspective, adapting to a broader range of standard summarization tasks. Throughout the knowledge usage in document summarization, from word-level knowledge [6] to document-level knowledge [7] and from internal knowledge [8] to external fact knowledge [9], we observe that various-formed knowledge appears and is incorporated for document summarization in different ways. In addition, empirical evidence from studies [10, 11, 12, 13] reported a worthwhile potentiality of leveraging different kinds of knowledge in both extractive and abstractive document summarization methods for single or multiple inputs. Also of note are the possibility and motivated envisagement of effectively blending a variety of knowledge for document summarization to enrich the fact and commonsense consistency of generated summaries. However, there is no existing work to summarize these research contributions. To fill this gap, we systematically investigate the knowledge and knowledge embedding methodologies under the document summarization view and report the results in this survey paper.

Comparisons to other surveys. The related works to our paper are surveys on knowledge, knowledge graphs, and knowledge embeddings in artificial

| Surveys | Coverage | Domain |
|------------------|-----------------|---------------|
| Wang et al. [14] | KGE | NLP |
| Cai et al. [15] | KG; GE | AI |
| Xu [16] | KG; GE | AI |
| Ji et al. [17] | KG; KGE | NLP |
| Hogan et al. [4] | KG; KGE | NLP |
| Ours | K; KE | DS |

Table 1: Outline of comparisons between existing relative surveys and ours. **K**, **G**, and **E** denote knowledge, graph, and embedding, respectively.

intelligence applications. Wang et al. [14] summarized specifically on knowledge graph embedding with a systematic review of existing embedding techniques in a range of the natural language processing applications, such as relation extraction and question answering. Cai et al. [15] proposed a classification of graph embedding work based on problem settings with descriptions of graph embedding techniques and applications in the artificial intelligence field, such as graph classification and graph visualization. Similarly, Xu [16] broadly categorized graph embedding methods according to base factors of graph embedding methods, such as matrix factorization, random walk, and neural network. Also, Xu [16] introduced representative real-world application examples from academia and industries in artificial intelligence. Ji et al. [17] provided a technical overview of knowledge graphs with knowledge graph embedding and introduced downstream knowledge-aware natural language processing applications such as question answering and recommendation systems. Hogan et al. [4] comprehensively introduced diverse concepts and aspects of knowledge graphs. Besides, Hogan et al. [4] distinguished open-source knowledge graphs as open knowledge graphs and regarded the internally constructed and utilized knowledge graphs as enterprise knowledge graphs. Table 1 presents an intuitive comparison between these relevant literature review articles and this paper. In a brief summary, the recent related works focus more on introducing either graph embeddings or knowledge graphs for a wide range of artificial intelligence applications in a general manner. Neither of them thoroughly targets a systematic view for one specific application. Differently, our survey studies a complete process of

leveraging the general knowledge in a promising natural language processing application, document summarization: from acquiring knowledge to embedding knowledge, followed by how embedding learning architectures generate and work with knowledge embeddings, under the document summarization view. We select, describe, and analyze the state-of-the-art document summarization works that embed general knowledge into models, and form the first systematic literature review of this kind.

Contributions of this survey. This survey contributes to the document summarization field with the investigation of the usage of knowledge and knowledge embeddings in document summarization models. Specifically, our *first contribution* is a taxonomy of the general knowledge leveraged in document summarization, presented in Section 2. In this paper, we consider all of the external relevant auxiliary information and the derived linguistic information in addition to the plain textual input as **knowledge** for document summarization, which is an expansion of the general factual knowledge. In our taxonomy, we broadly classify the knowledge incorporated in document summarization into four main categories: *native knowledge*, *linguistic knowledge*, *semantic knowledge*, and *topical knowledge*. The categorization is conducted lying on the layers of the knowledge from literalness to connotation, implied in hierarchies of the documentary information from word to the full text. The sub-categories are also discussed. Knowledge embeddings refer to low-dimensional and continuous representations of the knowledge [14], profiting better ways to permit various discrete-formed knowledge to be incorporated into learning models. Due to different-level knowledge leveraged for document summarization, a wide variety of knowledge embedding methodologies have been employed in document summarization models. Our *second contribution* is introducing a taxonomy of the existing knowledge embedding methodologies in document summarization tasks, introduced in Section 3 and how document summarization embedding learning architectures generate different knowledge embeddings as discussed in Section 4. Finally, we provide our envision about the future directions on the existing issues and unfilled gaps, which are aligned with the five goals of the

document summarization task in Section 5, forming our *third contribution*, followed by the conclusion in Section 6.

2. Knowledge Taxonomy

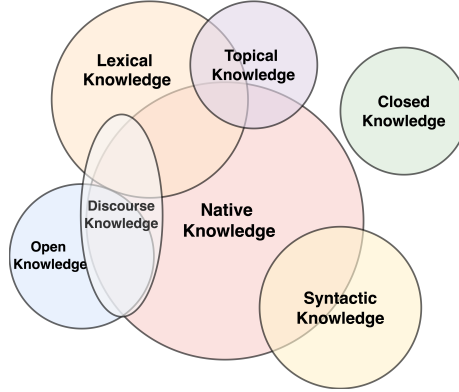


Figure 1: Knowledge categorization from the document summarization perspective.

In this survey, we classify the group of knowledge incorporated in document summarization models into four main categories. Relations among each type of knowledge are illustrated in Figure 1. Moreover, the investigation of knowledge leveraged in the state-of-the-art document summarization methods is shown in Table 2, with the order of the timeline of publication. It covers the usage status of knowledge graphs, the utilized knowledge relying on our proposed knowledge classification, and the main model architectures of those reviewed methods. The knowledge is obtained from the literal text or latent semantic space and can work alone or by merging to derive high-level information for the goals **G1** to **G5**.

2.1. Native knowledge

Native knowledge is the raw and plain textual data in the source text garnered without any filtration or transformation processing, such as the original words and sentences of the source text, typically leveraged as the auxiliary information [29]. This knowledge is captured in non-graph structures, such as in the form of token vectors, directly embedded into the model. It can represent

| Methods | DS | Knowledge | | | | | | | KG | Model |
|---------------------|-------|-----------|----|----|----|----|----|----|-----------------|-------------------|
| | Tasks | NK | LK | SK | DK | CK | OK | TK | | Architectures |
| A-SDS | | | | | | | | | | |
| ABS+AMR [18] | A/E | | | ✓ | | | | ✓ | ✓ | Encoder-Decoder |
| GAM+HBS [8] | | ✓ | ✓ | | | | | | ✓ | Encoder-Decoder |
| BERTSUM [19] | | ✓ | | | | | | | | BiTransformer |
| BERT+RL [20] | | ✓ | | | | | | | | Transformer |
| GraphWriter [21] | | ✓ | | | | | | ✓ | ✓ | G-Transformer |
| PG+PreTrained [22] | | ✓ | | | | | | | | Pointer Generator |
| IE+MSA [23] | | | | | | | | ✓ | ✓ | Encoder-Decoder |
| TXL+WikiKG [9] | | ✓ | | | | | | ✓ | ✓ | Transformer-XL |
| SemSUM [24] | | ✓ | | ✓ | | | | | ✓ | Transformer |
| GRF [25] | | ✓ | | | | | | ✓ | ✓ | GPT-2 |
| PGN+IDF [26] | | ✓ | | | | | | | | Encoder-Decoder |
| FASUM [27] | | ✓ | | | | | | ✓ | ✓ | Seq2Seq |
| SKGSUM [10] | | ✓ | ✓ | | ✓ | | ✓ | ✓ | Encoder-Decoder | |
| A-MDS | | | | | | | | | | |
| Seq2Seq+MTG [28] | S/M | | ✓ | | | | ✓ | ✓ | ✓ | Transformer |
| PEGASUS [29] | | ✓ | | | | | | | | Encoder-Decoder |
| ASGARD [30] | | ✓ | | | | | ✓ | ✓ | ✓ | Encoder-Decoder |
| BartGraphSumm [31] | | ✓ | ✓ | | | | ✓ | ✓ | ✓ | BART-Long |
| EMSUM [32] | | ✓ | ✓ | ✓ | | | | | ✓ | Transformer |
| BASS [12] | | ✓ | | ✓ | | | ✓ | ✓ | ✓ | Encoder-Decoder |
| E-SDS | | | | | | | | | | |
| FSGM [6] | | ✓ | ✓ | | | ✓ | | ✓ | ✓ | GraphRank |
| RNN+LSTM [33] | | ✓ | | | | | | | | Encoder-Decoder |
| HIBERT [34] | | ✓ | | | | | | | | Transformer |
| BERT+HGM [35] | | ✓ | | ✓ | | | | | ✓ | BERT+HGM |
| RST+spanBERT [36] | | ✓ | | | ✓ | | | | ✓ | Longformer |
| Topic-GraphSum [37] | | ✓ | | | | | | ✓ | ✓ | GNN |
| DISCOBERT [38] | | ✓ | ✓ | | ✓ | | | | ✓ | Transformer |
| kg-KMTR [11] | | ✓ | | | | | ✓ | | ✓ | TextRank+K-means |
| E-MDS | | | | | | | | | | |
| GRU+GCN [7] | | ✓ | ✓ | | ✓ | | | ✓ | ✓ | GNN |
| STDS [39] | | ✓ | | | | | | ✓ | | Encoder-Decoder |
| HETERSUMGRAPH [40] | | ✓ | ✓ | | | | | ✓ | | GNN |

Table 2: List of the representative **A**bstract or **E**xtractive **S**ingle- or **M**ulti-document summarization (**DS**) methods incorporated the knowledge, indicating the usage of knowledge graphs (**KG**) and the main model architectures. The described Native Knowledge (**NK**), Lexical Knowledge (**LK**), Syntactic Knowledge (**SK**), Discourse Knowledge (**DK**), Closed Knowledge (**CK**), Open Knowledge (**OK**), and Topical Knowledge (**TK**) are presented.

the maximal amount of the origin text information and enhance the content richness for promoting to achieve the goal **G1**.

2.2. Linguistic knowledge

Linguistic knowledge focuses on the source text information with an additional linguistic perspective than the native knowledge, such as the information related to the lexis, syntax, and grammar of the source text. Generally, linguistic knowledge is presented as in gauged relations among words or parsed relations of sentence dependencies.

Lexical knowledge. It is the estimated lexical relation knowledge among text entities (i.e., words), such as the centrality [41], textual similarity [7, 28, 42, 40, 10, 32, 13], semantic similarity [6, 8, 29], and salience [7, 8] information. This knowledge is in the form of numerical scores, infused as weights in learning models. It conduces to filter the relevant and salient text units for generating informative and succinct summaries, tallying with the goals **G1** and **G2**. Also, it captured word relations can enhance summary coherence for the goal **G3**.

Syntactic knowledge. It involves in syntactic dependency relations extracted by dependency parsers, such as the *JAMR* [43], *CoreNLP dependency parser* [44], and neural dependency parser [45]. This dependency relation forms the syntactic knowledge among words of each sentence, commonly modelled as dependency trees. The preserved syntactic relations can assist in determining redundant units and improving the summary coherence forward to the goals **G2** and **G3**.

Discourse knowledge. It covers discourse dependency relations concluded by discourse relation indicators via discovering deverbal noun references, event or entity continuations, discourse markers, or coreferent mentions [7, 42, 38], or gathered by discourse parsers [46]. This knowledge is usually formed as discourse graphs. It contains both syntactic and semantic information, excelling at redundancy recognition and logic enhancement, profiting the goals **G2** and **G3**.

2.3. Semantic knowledge

Semantic knowledge concentrates on conceptual and factual information gained from the real world or extracted from the source text. It is in the form of triplet (subject, predicate, object) and typically preserved in knowledge graph (KG) or knowledge base (KB).

Closed knowledge. It is the lexical relationship knowledge from the existing open-source and graph-based databases that contain general commonsense and human explicit knowledge, such as *WordNet* [47], *FrameNet* [48], *ConceptNet5* [49], and *Wikidata* [50]. This triplet-formed knowledge involves real-world facts that are surpassed to detect inconsistent fact errors in document summarization for achieving the goal **G5**.

Open knowledge. It is the ever-evolving and expansible knowledge of semantic relations extracted and accumulated from source corpora by information extraction tools, such as *Open-domain Information Extraction (OpenIE)* models [51, 52]. Similar to close knowledge, open knowledge is also in the form of the triplet (subject, predicate, object), but with more flexible subjects, predicates and objects. The semantic relations held by the open knowledge could also help improve the concision and logic of the summary, promoting the goal **G5**.

2.4. Topical knowledge

Topical knowledge is a latent knowledge of the source text, gained by topic models, such as the Latent Dirichlet Allocation (LDA) [53] or neural topic model (NTM) [54]. This knowledge mainly comprises the information of the topic salience [39] and topical relevance [37, 42]. It can indicate the phrase-level semantic information to enhance the summary coherence for the goal **G2** or imply the document-level semantic information for capturing relations among documents, benefiting the goal **G3**.

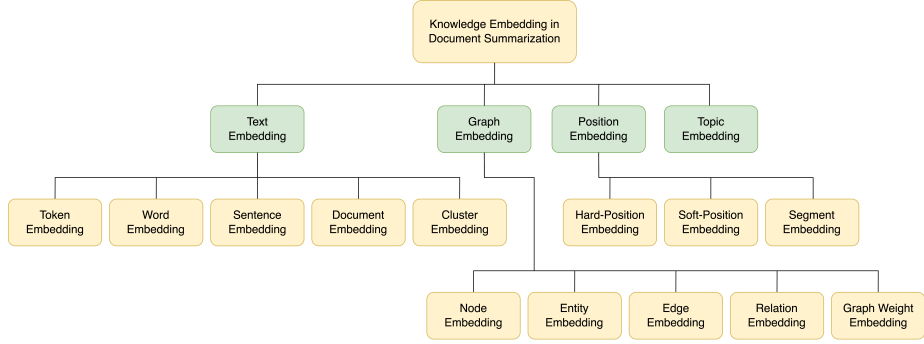


Figure 2: The taxonomy of knowledge embeddings utilized in document summarization tasks.

3. Knowledge Embedding Taxonomy

Native knowledge is usually in the original textual form, and its embedding relies on the embedding of the textual components in the document, such as token embedding, word embedding, sentence embedding and document embedding. *Linguistic knowledge* could be formed as texts or relations, and the latter is commonly modelled as a graph. Therefore, embedding linguistic knowledge covers both text embedding and graph embedding. *Semantic knowledge* similarly leverages both textual embedding and graph embedding. *Topical knowledge* is usually in data distribution form and requires to embed the distributions. In order to present the knowledge embedding applied in document summarization clearly, instead of grouping the embedding methods according to knowledge categorization, we propose **a new taxonomy** for knowledge embedding methods, as shown in Figure 2.

Text Embedding Many knowledge embedding methods in document summarization focus on using textual contents from the source corpus.

- *Token embedding* [9, 19, 25, 35, 30, 40, 55, 10, 31, 32] which is generally produced from input tokens by the last layer of the pretrained language model (e.g., Bidirectional Encoder Representations from Transformers (BERT)). The *WordPiece embedding* [20] is a special token embedding obtained by *WordPiece* tokenizers.

- *Word embedding* [6, 18, 8, 33, 23, 21, 28, 39, 20, 25, 24, 26] which is typically denoted as a vector of low dimension real numbers via methods, such as the one-hot vector and distributed representation. The *Word2Vec* is a general word embedding algorithm, producing the *Word2Vec embedding* [7, 22]. Also, the word vector garnered by the *Global Vectors For Word Representation (GloVe)* algorithm and the *FastText* mechanism is known as the *GloVe embedding* [33, 8, 40, 10] and the *FastText embedding* [22], respectively. Moreover, the *Context embedding* [18, 19] is a contextual vector for output words from the top layer of the language model, mapped with a weight matrix.
- *Sentence embedding* [8, 39, 11] which is typically a concatenation of word embeddings or gained by Sent2Vec [33]. In deep neural summarization methods, sentence embeddings are computed from word embeddings [7] or derived by language models (e.g., BERT). Besides, the TF-IDF is a general sentence embedding algorithm for the *TF-IDF embedding* [11]. The term frequency value is neglected in case the summary is formed by tremendously fewer tokens than the source document, known as the *IDF-weighted word embedding* [26]. Also, an *Elementary Discourse Unit (EDU)* is a sub-sentence phrase unit originating from RST discourse trees, represented by the *EDU embedding* [38]. The *Phrase embedding* [21] is a special sentence embedding produced from word embeddings run over last hidden states of neural networks (e.g., RNN). The *Title embedding* [21] is the title word embedding, regarding the title as a sentence, produced by neural networks (e.g., RNN) with last hidden states.
- *Document embedding* [7, 39, 20] which is the concatenation of sentence embeddings or computed from sentence embeddings by the neural model.
- *Cluster embedding* [7] which is resulted from averaging document embeddings, supplied in the form of real numbers.

Graph Embedding. Graph embedding methods can be applied on embedding

different components of the graph.

- *Node embedding* [56, 18, 24, 55, 27] which represents a graph node, computed by the network layer from aggregated local graph information of its adjacent nodes and relations. In terms of the node orientation, the node embedding can be further classified into *Forward-looking node embedding* and *Backward-looking node embedding* [21].
- *Entity embedding* [9, 32] which is a representation of a graph entity, learned from output vectors of pretrained language models or by techniques for modelling multi-relational data, such as the TransE [57].
- *Edge embedding* [18] which is the representation of an out-edge of the graph directed to the local parent node or global root node.
- *Relation embedding* [56, 9, 25, 55] which is the representation of a relationship or concept between entities, typically derived by TransE from the graph and known as the *Concept embedding* [55]. Besides, it can be captured by firstly aggregating node and edge embeddings and then transforming it via linear transformations followed by nonlinear activation functions (e.g., ReLU) [24]. Also, the relationship type can be indicated by the *Relation-type embedding* [25].
- *Graph weight embedding* [28] which is capable to represent the weight of both the node and edge of a graph, learned from the gating function or discretization of real numbers. A graph weight embedding that solely indicates the edge weight is represented as a token embedding and is equal to the number of merge operations increased by one, known as the *Edge weight embedding* [40, 32].

Topic Embedding. A *Topic embedding* [39, 37] is applied to embed topical information. It is a topic word vector typically composited from document embeddings or distilled subtopics. In deep learning architectures, it can be learned

by neural topic models. Besides, the *subtopic embedding* [39] is constructed by sentence embeddings.

Position Embedding. Position embedding is related to native knowledge. Its embedding is generated straightforward by using the token index information.

- *Hard-position embedding* [28, 19, 35, 31, 12] which is the numeric index of a token in its corresponding token sequence (i.e., sentence), also known as the *Positional embedding* [29, 25, 24].
- *Soft-position embedding* [55] which is the token index in a token sequence tree (i.e., sentence tree), represented as an integer number.
- *Segment embedding* [19, 35] which is a token notation assigned for discriminating multiple adjacent granularity levels (e.g., sentences) in a document, based on the parity of the level index.

4. Knowledge Embedding in Different Learning Architectures

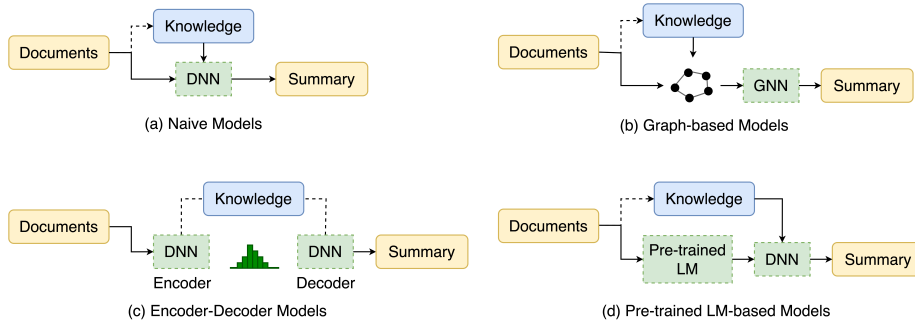


Figure 3: Learning architectures of embedding knowledge for document summarization. The dot line connected to knowledge indicates the knowledge could be obtained from the documents or external sources.

In this section, we discuss the reviewed works from the perspective of learning architectures applied for incorporating knowledge in document summarization models, with more attention to deep learning architectures. Figure 3 depicts

the four deep learning based architectures to embed knowledge into the summarizers. *Native models* (Figure 3 (a)) represent the works that only apply single Deep Neural Networks (DNN) for embedding knowledge. *Graph-based model* (Figure 3 (b)) utilize the graph-based neural networks to first form knowledge into graph, then learn from the graphs with graph neural networks (GNN), graph convolutional networks (GCN) or graph attention network (GAT). *Encoder-decoder models* (Figure 3 (c)) describe the works employ the encoder-decoder architecture and with different implementations for encoder and decoder. *Pre-trained Language model-based models* (Figure 3 (d)) introduce the works that are build upon the pre-trained representations. Additionally, we combine with the overview in previous sections and provide a summary for the reviewed works, as shown in Table 3, that covers main learning architectures for embeddings, types of document summarization tasks, and types of knowledge embeddings. We particularly highlight specific technique characteristics deriving knowledge embeddings for document summarization.

4.1. Naive Approaches

Models for document summarization can employ DNN directly for embedding the knowledge. Takase et al. [18] adopted structural syntactic and semantic information as knowledge and utilized a variant of child-sum Tree-LSTM to encode the syntactic and semantic information into fixed-length embedding. However, due to the advancement of deep learning architectures, in recent years, most works employ more complex architectures rather than simply utilizing only one deep neural network.

4.2. Graph-based Approaches

The *graph convolutional network* is a novel knowledge embedding approach in document summarization, majoring to embed graph-formed knowledge, such as the knowledge graph ConceptNet5 [25, 38]. As an upgrade from GCN, *graph attention network* is widely utilized in document summarization for embedding knowledge extracted by OpenIE or Stanford CoreNLP, preserved in graphs

| Learning Architectures | Methods | DS Tasks | Knowledge Embeddings |
|------------------------|---------------------|----------|-----------------------------|
| Transformer | SemSUM [24] | A - S | TOK, SEN, N, EDG, R, P |
| | Seq2Seq+MTG [28] | A - M | W, GW, P |
| | PEGASUS [29] | A - M | P |
| BART | BartGraphSumm [31] | A - M | TOK, P |
| BERT | BERTSUM [19] | A/E - S | TOK, W, CON, SEN, D, P, SEG |
| | HIBERT [34] | E - S | W, SEN, D, P |
| | BERT+HGM [35] | E - S | TOK, SEN, D, P, SEG |
| | RST+spanBERT [36] | E - S | SEN, EDU, D, N, ENT, R |
| | + WordPiece | A - S | WP, W, D |
| + GAT | Topic-GraphSum [37] | E - S | SEN, N, TOP |
| RoBERTa | ASGARD [30] | A - M | TOK, N |
| | EMSUM [32] | A - M | TOK, ENT, EW |
| | BASS [12] | A - S/M | TOK, N, P |
| RNN | GraphWriter [21] | A - S | W, SEN, TIT, N, ENT, R |
| + W2V | STDS [39] | E - M | W, W2V, SEN, D, TOP |
| + GCN | GRU+GCN [7] | E - M | W, W2V, SEN, D, CLU |
| LSTM | PGN+IDF [26] | A - S | W, IDF |
| + W2V / FT | PG+PreTrained [22] | A - S | W2V / FT |
| + GloVe | RNN+LSTM [33] | E - S | W, GV, SEN |
| TreeLSTM | ABS+AMR [18] | A - S | W, CON, N, EDG |
| GCN | GRF [25] | A - S | TOK, W, N, R, P |
| + SpanExt | DISCOBERT [38] | E - S | TOK, SEN, EDU |
| GAT | HETERSUMGRAPH [40] | E - S/M | W, GV, SEN, D, N, EW |
| | FASUM [27] | A - S | TOK, N |
| | + Glove | A - S | TOK, GV, SEN, TF-IDF, N |
| Word Rep. | IE+MSA [23] | A - S | CON, ENT, R |
| | FSGM [6] | E - S | W, SEN, D |
| + Glove | GAM+HBS [8] | A - S | W, GV, SEN, D |
| TransE | TXL+WikiKG [9] | A - S | TOK, ENT, R |
| TF - IDF | kg-KMTR [11] | E - S | SEN, TF-IDF |

Table 3: List of the representative **A**bstract or **E**xtractive **S**ingle- or **M**ulti-document summarization (**DS**) methods incorporated the knowledge, indicating knowledge embedding learning architectures. Embedding kinds of **T**OKEN, **W**ordPiece (**WP**), **W**ord, **W**ord2Vec (**W2V**), **G**loVe (**GV**), **F**astText (**FT**), **C**ONtext, **S**ENtence, **TF-IDF**, **IDF**, **E**DU, **T**ITle, **D**ocument, **C**LUsTer, **N**ode, **E**NTity, **E**DGe, **R**elation, **G**raph Weight (**GW**), **E**dge Weight (**EW**), **T**OPic, **P**osition, and **S**EGment are presented.

[27, 10]. In addition, Wang et al. [40] considered heterogeneous word-sentence relations to preserve hierarchical information as knowledge. The model treats the whole document as a graph and uses a graph attention network to learn the embedding. The knowledge-embedded sentences representations of sentence nodes is finally used for summary selection. Moreover, GraphWriter [21] employs the *Science-domain Information Extraction (SciIE)* for extracting science knowledge and embedding the knowledge by graph attention network.

4.3. Encoder-Decoder based Approaches

In this category, the encoder-decoder architecture is adopted for document summarization. The knowledge could be either embedded in encoder or decoder or both. EMSUM [32] uses the RoBERTa to embed the documentary information extracted by the *Coreference Resolution Tool* from *AllenNLP*. The Transformer-based encoder-decoder framework with a heterogeneous graph consisting of text units and entities as nodes. Zhu et al. [27] applied a Transformer-based encoder-decoder architecture via attention. The knowledge graph is obtained from information extraction results and participates in the decoder’s attention. Zheng et al. [39] adopted a bidirectional RNN encoder-decoder framework to learn sentence embedding for leveraging topic knowledge. Topic embedding is learnt from a soft-clustering on the sentence embedding and is fused with the model by non-linear transformation to the encoder. SemSUM [24] employs the Transformer encoder in a Transformer-based encoder-decoder model to encode knowledge and learn knowledge embeddings from the syntactic knowledge extracted by an off-the-shelf dependency parser. Wu et al. [12] utilized the semantic graph via dependency parsing and encoded knowledge in both encoder and decoder in a Transformer-based encoder-decoder model. Ji and Zhao [10] experimented with three types of knowledge constructing three knowledge graphs: entity graph, similarity graph, and discourse graph, and encoded the knowledge graph in both encoder and decoder of a Transformer-based encoder-decoder architecture. The entity graph is obtained by applying *Named Entity Recognition (NER)* and open information extraction via third part tools. The

similarity graph presents the sentence cosine similarity by using TF-IDF vectorization. The discourse graph follows the way to build an *Approximate Discourse Graph (ADG)* [7].

4.4. Pre-trained Model-based Approaches

Yuan et al. [35] obtained informative sentence representations on BERT, containing a hierarchical graph that brings the tokens at each granularity level be able to capture semantics from different sources. Xu et al. [38] leveraged discourse knowledge within structural discourse graphs constructed based on RST trees and coreference mentions. In addition, the work encodes the discourse graph using graph convolutional networks, which serve as graph encoders based upon sentence representations from BERT. The encoder finally outputs the knowledge incorporated embedding into MLP layers. Cui et al. [37] utilized topic knowledge via neural topic model and built a heterogeneous document graph consisting of sentence and topic nodes to learn the representations by a modified graph attention network with BERT. The representations of sentence nodes are extracted to compute the final summary. BartGraphSumm [31] is equipped with BART to encode the semantic knowledge extracted by the OpenIE from documents. Moreover, the model relies on the pre-trained RoBERTa encoder, which uses GAT to learn the graph extracted by OpenIE.

4.5. Non-deep learning Approaches.

In contrast to DNN-based approaches, non-deep learning approaches are commonly adopted for embedding external knowledge into document summarization. The traditional word vector representation method is still utilized in recent knowledge embedding approaches for document summarization [23]. Besides, advanced word vector representation methods, such as the distributed representation, are applied in both abstractive and extractive document summarization tasks, representing each word by its distributed representation [6, 8]. In addition, some document summarization knowledge embedding approaches adopt the linearization mechanism (e.g., TransE) to linearize the knowledge into

sequences for embedding into the model architecture [9]. Scarce approaches utilize straight the embedding algorithms, such as the TF-IDF algorithm, for learning knowledge embeddings in document summarization models [11].

5. Challenges and Future Opportunities

As still in its evolutionary stage, the research of embedding knowledge into document summarization faces numerous challenges and remains unfilled gaps. In this section, we discuss the challenges and promising avenues of ongoing and future works aligned with the goals **G1** to **G5**.

5.1. Knowledge Quality

For document summarization models that leverage knowledge, the knowledge that covers less information, retains fault information, or contains factual errors can significantly harm the summarization performance. There are some latent future directions to maintaining the quality of a knowledge base for document summarization to carry large amounts of essential and factually consistent information.

Knowledge Collection. The issue of fact coverage can occur due to the choice of information extraction strategies when collecting knowledge from texts. This issue may cause the lost of prominent information from source documents, thus degrading the quality of the generated summary [21]. Therefore, an effective extraction strategy designed for improving the coverage in knowledge collection is requisite to be explored, in order to reduce the missing knowledge in the distilling process. It can also further help produce informative summaries corresponding to the goals **G1** and **G2**. Besides, it is noted that determining the voice (i.e., active or passive) of sentences while extracting factual triples from the source text can advance the extracted information quality [58], which helps avoid missing information. More future research can be put in this direction to ensure the quality of collected knowledge.

Knowledge Purification. The knowledge graph entity disambiguation is to conquer entity ambiguity problems by matching ambiguous text entities to the corresponding knowledge graph entities for knowledge purification. However, when operating the disambiguation process to the knowledge graph to eliminate redundancy knowledge for document summarization, some of the salient text information could be lost [9]. Thus, a better strategy for knowledge graph disambiguation to condense the summary while remaining primary contents can be necessary. Also, more effective mechanisms for the entity recognition and entity linking within a knowledge graph to maintain more relations of knowledge entities are worth investigating [11]. Entity linking herein is the process of linking textual mentions of entities from source texts to the corresponding entities in a knowledge graph. These mechanisms can better reduce redundant information while retaining the knowledge base quality, achieving the goal **G2**.

Knowledge Consistency. Factual inconsistency errors refer to fact conflicts, categorized into contradicted fact (i.e., intrinsic error) and irrelevant fact (i.e., extrinsic error) to source text facts [59]. The knowledge from open-source knowledge graphs or extracted from the source corpora can inevitably involve varied intrinsic or extrinsic errors. Seriously erroneous knowledge can harm a knowledge-embedded summarizer’s performance terribly, mostly clashing with the goal **G5**. Even if factual inconsistency errors have been recognized and attached importance, scarce studies are qualified to precisely address and tackle inconsistency errors in document summarization. It is because inconsistency errors can be detected hardly by linguistic analysis. And since the knowledge databases for document summarization tasks are generally large-scale, it can cost laborious efforts to check each knowledge entity relation. Thus, exploring advanced ways to scan and solve inconsistent errors for incorporated external or personalized knowledge graphs, which contain plenty of mixed facts, can be a valuable future direction for knowledge-based document summarization.

5.2. Knowledge from Multi-facets

Except for investigations of manners to retain most of the superior knowledge, fusing knowledge from multiple facets to enhance the knowledge coverage in terms of different aspects can also promote the goals **G1** to **G5** for document summarization tasks.

Knowledge from Multiple Resources. Incorporating multiple types of knowledge from the real world and source corpora can be more beneficial in gathering more facts and prominent information in document summarization. It can improve the fact consistency of generated informative summaries, achieving the goals **G1** and **G5**. As aforementioned in Section 2, recent document summarization studies reported the possibility and advantages of blending different kinds of knowledge, expanding the knowledge base and enhancing the commonsense uniformity. Also, integrating knowledge graphs with abstract meaning representations to combine knowledge for document summarization can be an appealing research direction [24]. However, empirically verifying the efficiency of leveraging fused knowledge and exploring effective multiple knowledge combinations are not well-attended research areas.

Knowledge from Multiple Levels. Recent works reported that sentence-level or paragraph-level relation extraction methods might lose global relationship information from the entire document context [60]. The lack of high-level relations, e.g., relations among paragraphs, can mainly reduce the summary coherence and harm the goal **G3**. However, most of current knowledge extraction methods for document summarization still focus on firstly splitting the entire document into sentences and then extracting triples from the sentence span as the sentence-level relation extraction, which loses varying degrees of the context information and high-level knowledge. Thus, it inspires the research direction of extracting document-level knowledge for document summarization. Moreover, employing the novel text-to-graph summarizer with the knowledge graph usage to capture more multi-level relations among knowledge can be a promising research direction in document summarization [61].

5.3. Knowledge Embedding Techniques

As summarized in Section 4, the majority of recent document summarization works utilized encoder-decoder models, neural networks, and non-deep learning knowledge embedding methods to generate embeddings when incorporating the knowledge into models. Explorations and experiments of employing novel knowledge embedding approaches, which widely benefit natural language processing tasks not limited to document summarization tasks, to achieve the goals **G1** to **G5** can be worthwhile future works.

Novel Knowledge Embedding Methods. The *FocusE* [62] enhances the knowledge preservation by forming and merging textual information in numerical forms with lexical knowledge while remaining the textual information. It presents the potential of combining multiple forms of knowledge within one knowledge graph for further better knowledge embedding, improving the fact richness and accuracy for the goals **G1** to **G5**. Future research could adopt the idea and explore the ways to jointly embed knowledge from heterogeneous sources and to different forms.

Novel Learning Strategies. As discussed in Section 4, the mostly used learning architectures in the reviewed works are graph-based, encode-decoder and pre-trained models. There is a space to explore more learning architectures that could help achieve the goals **G1** to **G5** by incorporating knowledge. Reinforcement learning has been applied in document summarization as a part of the overall model to train the model by giving rewards. However, most of works utilize the evaluation metrics as the rewards. Few works consider the knowledge as part of the rewards [63]. Future works could investigate how to form the informative knowledge as rewards to train the summarizers. Besides, the recently popular *Prompting* [64] strategy promotes another way of adopting pretrained language models. It has not yet been well-adopted in summarization community. Exploring the methods of embedding knowledge for document summarization via prompting strategy is another promising future direction.

6. Conclusion

Along with the pursuit of more informative and coherent summaries with factual consistency, attention to knowledge embedding as an incorporation module for document summarizers to enhance model performance and improve summary quality gathered pace. In this paper, we surveyed the state-of-the-art approaches to embedding knowledge into document summarization models. To explicitly review each representative knowledge embedding approach in document summarization, we proposed taxonomies for knowledge and knowledge embeddings and explored embedding learning architectures under the document summarization perspective. Furthermore, we discussed open questions and appealing research directions for embedding knowledge in document summarization tasks, which we hope can drive new improvements in the document summarization field.

References

- [1] C. Ma, W. E. Zhang, M. Guo, H. Wang, Q. Z. Sheng, Multi-document Summarization via Deep Learning Techniques: A Survey, *ACM Computing Surveys* (2022) 1–35. doi:10.1145/3529754.
- [2] W. S. El-Kassas, C. R. Salama, A. A. Rafea, H. K. Mohamed, Automatic text summarization: A comprehensive survey, *ESWA* 165 (2021) 113679. doi:10.1016/j.eswa.2020.113679.
- [3] R. Ferreira, L. de Souza Cabral, F. L. G. de Freitas, R. D. Lins, G. P. e Silva, S. J. Simske, L. Favaro, A multi-document summarization system based on statistics and linguistic treatment, *ESWA* 41 (2014) 5780–5787. doi:10.1016/j.eswa.2014.03.023.
- [4] A. Hogan, E. Blomqvist, M. Cochez, C. d’Amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann, Knowledge graphs, *ACM Computing Surveys* 54 (2022) 1–37. doi:10.1145/3447772.

- [5] S. Gao, X. Chen, Z. Ren, D. Zhao, R. Yan, From Standard Summarization to New Tasks and Beyond: Summarization with Manifold Information, in: IJCAI, 2021, pp. 4854–4860. doi:10.48550/arXiv.2005.04684.
- [6] X. Han, T. Lv, Z. Hu, X. Wang, C. Wang, Text Summarization Using FrameNet-Based Semantic Graph Model, Scientific Programming 2016 (2016) 1–10. doi:10.1155/2016/5130603.
- [7] M. Yasunaga, R. Zhang, K. Meelu, A. Pareek, K. Srinivasan, D. Radev, Graph-based Neural Multi-Document Summarization, in: CoNLL, 2017, pp. 452–462. doi:10.18653/v1/K17-1045.
- [8] J. Tan, X. Wan, J. Xiao, Abstractive Document Summarization with a Graph-Based Attentional Neural Model, in: ACL, 2017, pp. 1171–1181. doi:10.18653/v1/P17-1108.
- [9] B. Gunel, C. Zhu, M. Zeng, X. Huang, Mind The Facts: Knowledge-Boosted Coherent Abstractive Text Summarization, in: NeurIPS, 2019, pp. 1–7. doi:10.48550/arXiv.2006.15435.
- [10] X. Ji, W. Zhao, SKGSUM: Abstractive Document Summarization with Semantic Knowledge Graphs, in: IJCNN, 2021, pp. 1–8. doi:10.1109/IJCNN52387.2021.9533494.
- [11] T. Tang, T. Yuan, X. Tang, D. Chen, Incorporating External Knowledge into Unsupervised Graph Model for Document Summarization, Electronics 9 (2020) 1–13. doi:10.3390/electronics9091520.
- [12] W. Wu, W. Li, X. Xiao, J. Liu, Z. Cao, S. Li, H. Wu, H. Wang, BASS: Boosting Abstractive Summarization with Unified Semantic Graph, in: ACL-IJCNLP, 2021, pp. 6052–6067. doi:10.18653/v1/2021.acl-long.472.
- [13] M. Chen, W. Li, J. Liu, X. Xiao, H. Wu, H. Wang, SgSum: Transforming Multi-document Summarization into Sub-graph Selection, in: EMNLP, 2021, pp. 4063–4074. doi:10.18653/v1/2021.emnlp-main.333.

- [14] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge Graph Embedding: A Survey of Approaches and Applications, *IEEE TKDE* 29 (2017) 2724–2743. doi:10.1109/TKDE.2017.2754499.
- [15] H. Cai, V. W. Zheng, K. C.-C. Chang, A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications, *IEEE TKDE* 30 (2018) 1616–1637. doi:10.1109/TKDE.2018.2807452.
- [16] M. Xu, Understanding Graph Embedding Methods and Their Applications, *SIREV* 63 (2021) 825–853. doi:10.1137/20M1386062.
- [17] S. Ji, S. Pan, E. Cambria, P. Marttinen, P. S. Yu, A Survey on Knowledge Graphs: Representation, Acquisition and Applications, *IEEE TNNLS* 33 (2022) 494–514. doi:10.1109/TNNLS.2021.3070843.
- [18] S. Takase, J. Suzuki, N. Okazaki, T. Hirao, M. Nagata, Neural Headline Generation on Abstract Meaning Representation, in: *EMNLP*, 2016, pp. 1054–1059. doi:10.18653/v1/D16-1112.
- [19] Y. Liu, M. Lapata, Text Summarization with Pretrained Encoders, in: *EMNLP-IJCNLP*, 2019, pp. 3730–3740. doi:10.18653/v1/D19-1387.
- [20] H. Zhang, J. Cai, J. Xu, J. Wang, Pretraining-Based Natural Language Generation for Text Summarization, in: *CoNLL*, 2019, pp. 789–797. doi:10.18653/v1/K19-1074.
- [21] R. Koncel-Kedziorski, D. Bekal, Y. Luan, M. Lapata, H. Hajishirzi, Text Generation from Knowledge Graphs with Graph Transformers, in: *NAACL-HLT*, 2019, pp. 2284–2293. doi:10.18653/v1/N19-1238.
- [22] D. T. Anh, N. T. T. Trang, Abstractive Text Summarization Using Pointer-Generator Networks With Pre-Trained Word Embedding, in: *SoICT*, 2019, pp. 473–478. doi:10.1145/3368926.3369728.
- [23] J. Guan, Y. Wang, M. Huang, Story Ending Generation with Incremental Encoding and Commonsense Knowledge, in: *AAAI-IAAI-EAAI*, 2019, pp. 6473–6480. doi:10.1609/aaai.v33i01.33016473.

- [24] H. Jin, T. Wang, X. Wan, SemSUM: Semantic Dependency Guided Neural Abstractive Summarization, in: AAAI, 2020, pp. 8026–8033. doi:10.1609/aaai.v34i05.6312.
- [25] H. Ji, P. Ke, S. Huang, F. Wei, X. Zhu, M. Huang, Language Generation with Multi-Hop Reasoning on Commonsense Knowledge Graph, in: EMNLP, 2020, pp. 725–736. doi:10.18653/v1/2020.emnlp-main.54.
- [26] J. You, C. Hu, H. Kamigaito, H. Takamura, M. Okumura, Abstractive Document Summarization with Word Embedding Reconstruction, in: RANLP, 2021, pp. 1586–1596. doi:10.26615/978-954-452-072-4_178.
- [27] C. Zhu, W. Hinthorn, R. Xu, Q. Zeng, M. Zeng, X. Huang, M. Jiang, Enhancing Factual Consistency of Abstractive Summarization, in: NAACL-HLT, 2021, pp. 718–733. doi:10.18653/v1/2021.naacl-main.58.
- [28] A. Fan, C. Gardent, C. Braud, A. Bordes, Using Local Knowledge Graph Construction to Scale Seq2Seq Models to Multi-Document Inputs, in: EMNLP-IJCNLP, 2019, pp. 4186–4196. doi:10.18653/v1/D19-1428.
- [29] J. Zhang, Y. Zhao, M. Saleh, P. J. Liu, PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, in: ICML, 2020, pp. 11328–11339. doi:10.48550/arXiv.1912.08777.
- [30] L. Huang, L. Wu, L. Wang, Knowledge Graph-Augmented Abstractive Summarization with Semantic-Driven Cloze Reward, in: ACL, 2020, pp. 5094–5107. doi:10.18653/v1/2020.acl-main.457.
- [31] R. Pasunuru, M. Liu, M. Bansal, S. Ravi, M. Dreyer, Efficiently Summarizing Text and Graph Encodings of Multi-Document Clusters, in: NAACL-HLT, 2021, pp. 4768–4779. doi:10.18653/v1/2021.naacl-main.380.
- [32] H. Zhou, W. Ren, G. Liu, B. Su, W. Lu, Entity-Aware Abstractive Multi-Document Summarization, in: ACL-IJCNLP, 2021, pp. 351–362. doi:10.18653/v1/2021.findings-acl.30.

- [33] C. Zhang, S. Sah, T. Nguyen, D. K. Peri, A. C. Loui, C. Salvaggio, R. W. Ptucha, Semantic Sentence Embeddings for Paraphrasing and Text Summarization, *IEEE GlobalSIP* (2017) 705–709. doi:10.1109/GlobalSIP.2017.8309051.
- [34] X. Zhang, F. Wei, M. Zhou, HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization, in: *ACL*, 2019, pp. 5059–5069. doi:10.18653/v1/P19-1499.
- [35] R. Yuan, Z. Wang, W. Li, Fact-level Extractive Summarization with Hierarchical Graph Mask on BERT, in: *COLING*, 2020, pp. 5629–5639. doi:10.18653/v1/2020.coling-main.493.
- [36] Y. J. Huang, S. Kurohashi, Extractive Summarization Considering Discourse and Coreference Relations based on Heterogeneous Graph, in: *EACL*, 2021, pp. 3046–3052. doi:10.18653/v1/2021.eacl-main.265.
- [37] P. Cui, L. Hu, Y. Liu, Enhancing Extractive Text Summarization with Topic-Aware Graph Neural Networks, in: *COLING*, 2020, pp. 5360–5371. doi:10.18653/v1/2020.coling-main.468.
- [38] J. Xu, Z. Gan, Y. Cheng, J. Liu, Discourse-Aware Neural Extractive Text Summarization, in: *ACL*, 2020, pp. 5021–5031. doi:10.18653/v1/2020.acl-main.451.
- [39] X. Zheng, A. Sun, J. Li, K. Muthuswamy, Subtopic-driven Multi-Document Summarization, in: *EMNLP-IJCNLP*, 2019, pp. 3153–3162. doi:10.18653/v1/D19-1311.
- [40] D. Wang, P. Liu, Y. Zheng, X. Qiu, X. Huang, Heterogeneous Graph Neural Networks for Extractive Document Summarization, in: *ACL*, 2020, pp. 6209–6219. doi:10.18653/v1/2020.acl-main.553.
- [41] G. Erkan, D. R. Radev, LexRank: Graph-Based Lexical Centrality as Salience in Text Summarization, *Journal of Artificial Intelligence Research* 22 (2004) 457–479. doi:10.48550/arXiv.1109.2128.

- [42] W. Li, X. Xiao, J. Liu, H. Wu, H. Wang, J. Du, Leveraging Graph to Improve Abstractive Multi-Document Summarization, in: ACL, 2020, pp. 6232–6243. doi:10.18653/v1/2020.acl-main.555.
- [43] J. Flanigan, S. Thomson, J. Carbonell, C. Dyer, N. A. Smith, A Discriminative Graph-Based Parser for the Abstract Meaning Representation, in: ACL, 2014, pp. 1426–1436. doi:10.3115/v1/P14-1134.
- [44] K. M. Hermann, T. Kočiský, E. Grefenstette, L. Espeholt, W. Kay, M. Su-leyman, P. Blunsom, Teaching Machines to Read and Comprehend, in: NeurIPS, 2015, pp. 1693–1701. doi:10.48550/arXiv.1506.03340.
- [45] T. Dozat, C. D. Manning, Deep Biaffine Attention for Neural Dependency Parsing, in: ICLR, 2017, pp. 1–8. doi:10.48550/arXiv.1611.01734.
- [46] Y. Ji, J. Eisenstein, Representation Learning for Text-level Discourse Parsing, in: ACL, 2014, pp. 13–24. doi:10.3115/v1/P14-1002.
- [47] G. A. Miller, WordNet: A Lexical Database for English, Communications of the ACM 38 (1995) 39–41. doi:10.1145/219717.219748.
- [48] J. Ruppenhofer, M. Ellsworth, M. R. L. Petruck, C. R. Johnson, J. Schef-fczyk, FrameNet II: Extended theory and practice, FrameNet Project (2006).
- [49] R. Speer, C. Havasi, Representing General Relational Knowledge in ConceptNet 5, in: LREC, 2012, pp. 3679–3686.
- [50] D. Vrandečić, M. Krötzsch, Wikidata: A Free Collaborative Knowledge-base, Communications of the ACM 57 (2014) 78–85. doi:10.1145/2629489.
- [51] G. Angeli, M. J. J. Premkumar, C. D. Manning, Leveraging Linguistic Structure For Open Domain Information Extraction, in: ACL-IJCNLP, 2015, pp. 344–354. doi:10.3115/v1/P15-1034.

- [52] G. Stanovsky, J. Michael, L. Zettlemoyer, I. Dagan, Supervised Open Information Extraction, in: NAACL-HLT, 2018, pp. 885–895. doi:10.18653/v1/N18-1081.
- [53] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent Dirichlet Allocation, Journal of Machine Learning Research 3 (2003) 993–1022.
- [54] Y. Miao, E. Grefenstette, P. Blunsom, Discovering Discrete Latent Topics with Neural Variational Inference, in: ICML, volume 70, 2017, pp. 2410–2419. doi:10.48550/arXiv.1706.00359.
- [55] Y. Liu, Y. Wan, L. He, H. Peng, P. S. Yu, KG-BART: Knowledge Graph-Augmented BART for Generative Commonsense Reasoning, in: AAAI, volume 35, 2021, pp. 6418–6425. doi:10.48550/arXiv.2009.12677.
- [56] F. Liu, J. Flanigan, S. Thomson, N. Sadeh, N. A. Smith, Toward Abstractive Summarization Using Semantic Representations, in: NAACL-HLT, 2015, pp. 1077–1086. doi:10.3115/v1/N15-1114.
- [57] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston, O. Yakhnenko, Translating Embeddings for Modeling Multi-Relational Data, in: NeurIPS, volume 26, 2013, pp. 2787–2795.
- [58] A. Abdi, N. Idris, R. M. Alguliyev, R. M. Aliguliyev, Query-based multi-documents summarization using linguistic knowledge and content word expansion, Soft Computing 21 (2017) 1785–1801. doi:10.1007/s00500-015-1881-4.
- [59] Y. Xie, F. Sun, Y. Deng, Y. Li, B. Ding, Factual Consistency Evaluation for Text Summarization via Counterfactual Estimation, in: EMNLP, 2021, pp. 100–110. doi:10.18653/v1/2021.findings-emnlp.10.
- [60] S. Jain, M. van Zuylen, H. Hajishirzi, I. Beltagy, SciREX: A Challenge Dataset for Document-Level Information Extraction, in: ACL, 2020, pp. 7506–7516. doi:10.18653/v1/2020.acl-main.670.

- [61] Z. Wu, R. Koncel-Kedziorski, M. Ostendorf, H. Hajishirzi, Extracting Summary Knowledge Graphs from Long Documents, arXiv abs/2009.09162 (2020) 1–8. doi:10.48550/arXiv.2009.09162.
- [62] S. Pai, L. Costabello, Learning Embeddings from Knowledge Graphs With Numeric Edge Attributes, in: IJCAI, 2021, pp. 2869–2875. doi:10.24963/ijcai.2021/395.
- [63] E. Sharma, L. Huang, Z. Hu, L. Wang, An Entity-Driven Framework for Abstractive Summarization, in: EMNLP-IJCNLP, 2019, pp. 3280–3291. doi:10.18653/v1/D19-1323.
- [64] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing, arXiv preprint arXiv:2107.13586 (2021).