This article has been accepted for publication in IEEE Transactions on Knowledge and Data Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2023.3304385

A Survey on Neural Data-to-Text Generation

Yupian Lin, Tong Ruan, Jingping Liu, and Haofen Wang

Abstract—Data-to-text Generation (D2T) aims to generate textual natural language statements that can fluently and precisely describe the structured data such as graphs, tables, and meaning representations (MRs) in the form of key-value pairs. It is a typical and crucial task in natural language generation (NLG). Early D2T systems generated texts with the cost of human engineering in designing domain specific rules and templates, and achieved acceptable performance in coherence, fluency, and fidelity. In recent years, the data-driven D2T systems based on deep learning have reached state-of-the-art (SOTA) performance in more challenging datasets. In this paper, we provide a comprehensive review on existing neural data-to-text generation approaches. We first introduce available D2T resources, including systematically categorized D2T datasets and mainstream evaluation metrics. Next, we survey existing works based on the taxonomy along two axes: neural end-to-end D2T and neural modular D2T. We also discuss the potential applications and the adverse impacts. Finally, we present readers with the challenges faced by neural D2T and outline some potential future directions in this area.

Index Terms—Natural language processing, natural language generation, data-to-text generation, survey, deep learning

1 Introduction

Data-to-Text Generation refers to the task of generating natural language from structured data such as graphs, tables, and meaning representations. D2T is a classic and important task in natural language generation. It plays an essential role in human-computer interaction systems, since natural language that accurately and fluently describes structured data assists people in understanding data efficiently and easily.

Evolution of D2T. In the initial development period of the data-to-text generation, the traditional modular approaches that generate text based on heuristic rules or templates have occupied a dominant position. These traditional modular approaches follow the pipeline architecture for natural language generation. In 1992, McKeown [1] divided the pipeline architecture into three stages, namely, document planning, micro planning, and surface realization. In 1997, Reiter and Dale [2] further subdivided the pipeline architecture into six modules, namely, (1) content determination, (2) discourse planning, (3) sentence aggregation, (4) lexicalization, (5) referring expression generation, and (6) linguistic realization.

However, these heuristic-driven approaches are not scalable or adaptable to new domains, paving the way for statistical approaches based on probabilistic language generation process [3], probabilistic context-free grammar [4], and others [5], [6], [7]. In 2013, word vectors [8] marked the beginning of the era of language modeling based on neural networks, and promoted the development of sequence-to-sequence generation tasks such as Machine Translation [9] [10], Text Summarization [11] and Dialogue [12]. Benefiting from the success of the sequence generation tasks, the performance of D2T has been greatly improved by using the neural end-to-end framework. Different from the traditional modular approaches, the neural end-to-end D2T models [13], [14], [15], [16], [17] directly modeled the corresponding relationship between the structured data and the reference text, avoiding the risk of error propagation. Although the existing neural end-to-end D2T models can generate fluent texts, there are still many problems, such as weak controllability, poor faithfulness, and low coverage. Thus, the neural modular approaches, including the two-stage approaches [18], [19], [20], [21] and neural template-based approaches [22], [23], [24], [25], were explored by more researchers in order to find a better balance between controllability and efficiency. Specifically, the two-stage approaches decoupled the content planning from neural end-to-end framework to improve the controllability and faithfulness of generated text. While the neural template-based approaches aimed to learn latent and discrete templates from training data.

Motivations of This Survey. In 2017, the publication of the WebNLG [26], RotoWire [27], and E2E [28] datasets officially ushered in the era of data-driven neural D2T. Over the past few years, a considerable number of studies have applied deep learning to D2T and successively advanced the SOTA performance. On the other hand, although D2T studies have been flourishing for many years, to the best of our knowledge, there are few reviews in this field so far [29]. Therefore, we conduct a up-to-date survey on neural D2T in order to enlighten and guide researchers and practitioners.

Contributions of This Survey. We give an up-to-date synthesis of the neural D2T researches, as well as the various deep learning technologies and architectures. To offer constructive resources for D2T research community, we consolidate existing D2T datasets, and mainstream evaluation metrics. Next, we give a detailed and thorough taxonomy survey based on two axes: neural end-to-end D2T and neural modular D2T. We also discuss the potential applications and the adverse impacts of neural D2T. Finally, we present readers with the difficulties D2T systems confront, and then highlight future directions in this field.

- *Yupian Lin, Tong Ruan, and Jingping Liu were with the School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China.*
E-mail: yupianlin@aliyun.com, {ruantong, jingpingliu}@ecust.edu.cn.
Corresponding author: Tong Ruan.
- *Haofen Wang was with the College of Design and Innovation, Tongji University, Shanghai 200092, China.*
E-mail: carter.whfcarter@gmail.com

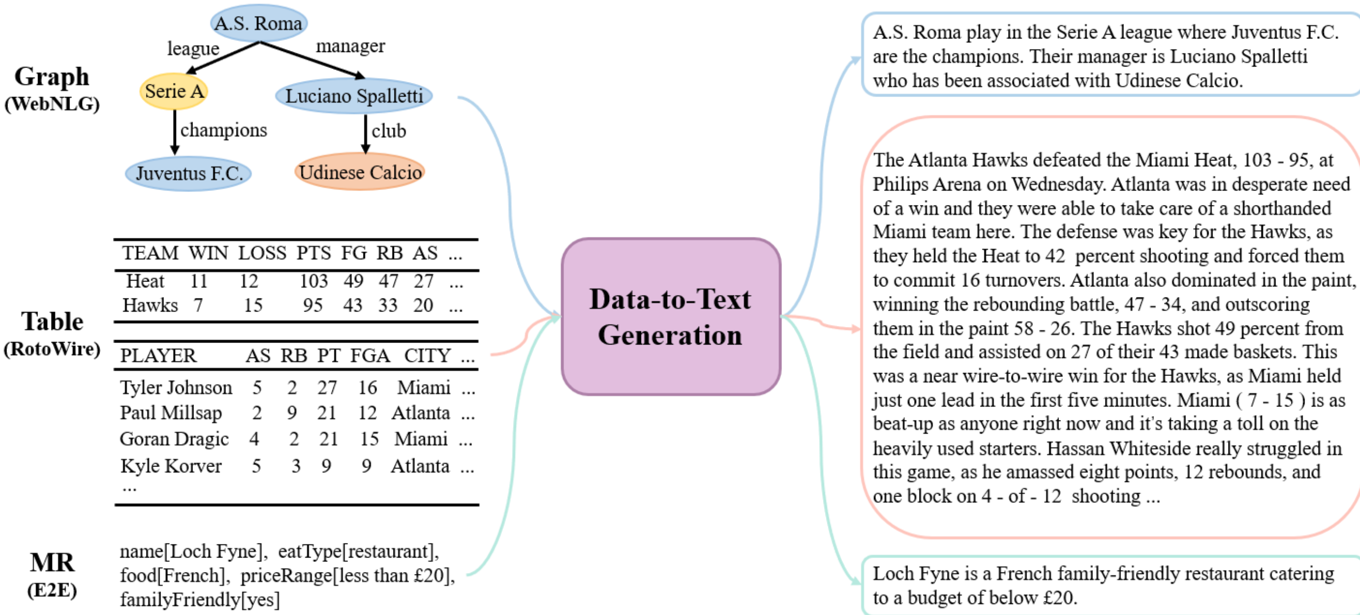

Fig. 1. The illustration of D2T from three types of structured data. Top: Graph (WebNLG [26]). Middle: Table (RotoWire [27]). Bottom: MR (E2E [28]).

2 Background

At the first, we give a formal formulation of the data-to-text generation task. We then introduce the widely-used D2T datasets from three types of structured data. Next, we detail the evaluation metrics and summarize the traditional approaches to D2T.

2.1 What is D2T?

Data-to-text generation aims to generate the textual output that can accurately and fluently describe the non-linguistic structured data. Structured data is usually stored in different forms, including graphs, tables, and MRs. Figure 1 shows examples of data-to-text generation from above types of structured data.

There are two main problems to be solved in the task of data-to-text generation: ”What to say” and ”How to say”. Specifically, ”What to say” means to analyze and filter given structured data, from which all or part of the data is selected for abstraction and association. ”How to say” refers to accurately and fluently describe the selected data through natural language.

2.2 Datasets

At present, the datasets of data-to-text generation tasks are mainly divided into three categories according to the types of structured data: (1) graph-to-text generation, (2) table-to-text generation, and (3) MR-to-text generation. Table 1 lists the widely-used datasets for D2T.

2.2.1 Graph-to-Text Generation

Since the commonly used graph data mainly include Knowledge Graph (KG) and Abstract Meaning Representation (AMR) , graph-to-text generation can be further subdivided into KG-to-text generation and AMR-to-text generation.

KG-to-Text Generation. WebNLG [26], released in 2017, is a very classic dataset that contains sets of RDF triples extracted from DBPedia. Later Enriched WebNLG [30] and WebNLG+ [31] are constructed by adding German and Russian corpora respectively on this basic WebNLG dataset and enriching the resources of the modular architecture. Similarly, the AGENDA [32] and WITA [33] are constructed by extracting KB triples from the target text. But DART [34] dataset is built by extracting triples from tables. KGTEXT [35] dataset is constructed by using Wikipedia hyperlinks to query the knowledge graph corresponding to the text. EventNarrative [36] is a large-scale event-centric dataset which is created by using matching algorithms to match KG to text automatically. In addition, GenWiki [37] contains 1.3 million non-parallel text and graphs with shared content for training unsupervised KG-to-text generation models. TEKGEN [38] dataset is created using distant supervision by aligning Wikidata triples to Wikipedia text.

AMR-to-Text Generation. AMR is a syntactic independent semantic representation of sentences. It can represent sentences as single-root directed acyclic graphs according to semantic structure. Thus, AMR is well suited for semantically related NLP tasks. In AMR-to-text Generation, currently commonly used public datasets include AMR15 (LDC2015E86), AMR17 (LDC2017T10) [39] and AMR20 (LDC2020T02) [40]. These datasets are human-annotated treebanks, which contain 19572, 39260, and 59255 cross-domain natural language texts respectively. Each sample contains a sentence and a corresponding AMR diagram.

2.2.2 Table-to-Text Generation

According to whether the data belongs to a specific domain, the table-to-text generation can be subdivided into the following two categories:

Domain-specific Table-to-Text Generation. In the sports domain, RotoWire [27] consists of textual descriptions of

This article has been accepted for publication in IEEE Transactions on Knowledge and Data Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2023.3304385

TABLE 1
List of datasets for D2T. "Avg.Len" refers to the average length of samples. "Plan" refers to whether it contains the content planning.

Type	Sub.Type	Datasets	Domain	Language	Samples	Avg.Len	Plan	Year
Graph-to-text	KG-to-text	WebNLG	Wikipedia	English	9.6K	22.69	No	2017
		Enriched WebNLG	Wikipedia	English, German	32.9K	19.67	Yes	2018
		WebNLG+	Wikipedia	English, Russian	25.3K	20.57	Yes	2020
		AGENDA	Wikipedia	English	40.7K	141.2	No	2019
		WITA	Wikipedia	English	55.4K	18.8	No	2020
		DART	Wikipedia	English	82.2K	21.6	No	2021
		KGTEXT	Wikipedia	English	16M	20.2	No	2020
		GenWiki	Wikipedia	English	1336.7K	21.46	No	2020
		EventNarrative	Wikipedia (EventKG)	English	224.4K	50.58	No	2021
		TEKGEN	Wikipedia	English	5723K	21.2	No	2021
	AMR-to-text	AMR15	Discussion forum etc.	English	19.5K	21.3	No	2015
		AMR17	Discussion forum etc.	English	39.2K	20.4	No	2017
		AMR20	Discussion forum etc.	English	59.2K	16.9	No	2020
Table-to-Text	Domain-specific	RotoWire	Sports	English	4.9K	337.1	No	2017
		RotoWire-Modified	Sports	English	3.7K	384	No	2019
		RotoWire-FG	Sports	English	7.5K	205.9	No	2020
		ESPN	Sports	English	15.0K	9.5	No	2018
		MLB	Sports	English	26.3K	542.05	No	2019
		BioLeaflets	Biomedical	English	77.1K	412.9	No	2021
		numericNLG	Scientific	English	1.3K	94	No	2021
	Open-domain	WikiBio	Wikipedia	English	728.2K	26.1	No	2016
		WikiPerson	Wikipedia	English	311.5K	88.3	No	2018
		WikiTableText	Wikipedia	English	13K	13.91	No	2018
		WikiTablePara	Wikipedia	English	–	760	Yes	2018
		WikiTableT	Wikipedia	English	15K	115.9	No	2021
		Wiki3C	Wikipedia	English	10.2K	–	No	2021
		ToTTo	Wikipedia	English	136.0K	17.4	No	2020
		TabFact	Wikipedia	English	118K	13.8	No	2020
		TWT	Wikipedia	English	177.7K	16.4	No	2021
		LogicNLG	Wikipedia	English	37.0K	14.2	No	2020
		Logic2Text	Wikipedia (WikiTables)	English	10.7K	16.77	No	2020
MR-to-Text	Slot-value pairs	E2E	Restaurant	English	51.4K	22.41	No	2017
		Cleaned E2E	Restaurant	English	42.4K	22.9	No	2018
		Czech D2T	Restaurant	English, Czech	5.2K	–	No	2019
		ViGGO	Video Game	English	6.9K	25.01	No	2019
		KVRET	Multi-domain	English	3.0K	47.25	No	2017
		MultiWOZ	Multi-domain	English	10.4K	15.12	No	2018
	Attribute-value pairs	SUMTIME	Weather	English	1.2K	16.2	No	2008
		WEATHERGOV	Weather	English	22.1K	28.7	No	2009
		RoboCup	Sports	English	1.9K	5.7	No	2008
		CACAPO	Multi-domain	English, Dutch	21.0K	16.86	Yes	2020

basketball games and numerous statistical tables; Rotowire-Modified [41] is a more reliable version of Rotowire; the later version, Rotowire-FG [42], is constructed by using sophisticated heuristics to remove redundant numbers and irrelevant schedules. And ESPN [43] contains 15,054 NBA game result headlines during 2006-2017 from the ESPN website, paired with their corresponding game statistics. In addition, MLB [13] consists of richer baseball game scoring summaries and longer textual descriptions than RotoWire. In the biomedical domain, BioLeaflets [44] consists of packaging leaflets for approved medicines in Europe. The main challenges of BioLeaflets include multi-section target text, specialized medical vocabulary, and syntax. Moreover, the tables in numericNLG [45] are numerical tables. The difficulty lies in the need for numerical reasoning.

Open-domain Table-to-Text Generation. Open-domain table-to-text generation mainly develops in three aspects.

(1) *From short text to long text.* WikiBio [46] is made up of biographical tables along with the first sentence of each biography article from Wikipedia. Compared to WikiBio, WikiPerson [47] contains more sentences corresponding to the table. WikiTablePara [48] is constructed by filtering more than 2000 tables and finally retaining 171 tables, each of which has a four-paragraph reference description. The WikiTableT [49] is constructed by combining previous work and WikiTableText [50] to generate the large-scale dataset.

(2) *Controlled D2T for content fidelity.* Wiki3C [51] is designed for sensitivity around information fidelity hence demonstrating the high capability of divergences. ToTTo [52] includes tables with multiple patterns, and completes the text generation innovation of controlled cells in terms of content selection. In TabFact [53], each table corresponds to two hypotheses for studying fact verification. TWT [54] is constructed by repurposing ToTTo and TabFact, targeting controlled D2T. It provides prefix-target pairs to control the topic of generated text.

(3) *D2T with logical reasoning.* The significant difference between LogicNLG [55] and the above-mentioned open-domain datasets is that the former contains annotated statements with rich logical inferences. Similarly, Logic2text [56] consists of tables, corresponding texts, and logical forms that describe the semantics of the target text via diversified graphs. It not only simply describes the content of the table, but also summarizes the information in the table through logical reasoning such as count, superlative and comparative.

2.2.3 MR-to-Text Generation

In MR-to-text generation, the structured data is usually expressed in the form of key-value pairs. Specifically, the structured data in the dialogue system is expressed in the form of slot-value pairs (SVP); the structured data in scenarios where the data fields are relatively fixed, such as news broadcasts and weather broadcasts, is often expressed in attribute-value pairs (AVP).

SVP-to-Text Generation. E2E [28] consists of more than 50k combinations of a dialogue-act-based MR and 8.1 references on average. In order to reduce the influence of semantic noise matters, Cleaned E2E [57] is produced in 2018. Moreover, Czech D2T [58] is the first Czech dataset targeted at end-to-end D2T. ViGGO [59] contains approximately 7K pairs of MRs and reference utterances about over 100 video games. The KVRET [60] and MultiWOZ [61] are large-scale multi-domain datasets containing thousands of dialogues.

AVP-to-Text Generation. SUMTIME [62] is a small dataset for training the early weather forecasting report system, which only contains 1,220 examples. Besides, WEATHERGOV [63] is a larger dataset of 22,146 scenarios. RoboCup [64] contains 1,919 scenarios from the 2001–2004 Robocup finals. CACAPO [65] contains almost 21,052 examples from human-written news texts in the multi-domain, together with aligned attribute-value pairs. Moreover, CACAPO can be used to train not only end-to-end D2T models but also modular D2T models.

2.3 Evaluation Metrics

2.3.1 Automatic Evaluation Metrics

As shown in the table 2, current popular automatic evaluation metrics for D2T task can be roughly put into the following three categories: (1) lexical similarity; (2) semantic equivalence; (3) faithfulness.

(1) Lexical Similarity. Early traditional metrics measure lexical similarity by calculating the overlap of n-gram at the word level between the generated text and the reference text, such as BLEU [66], ROUGE [67], METEOR [68], NIST [69] and CIDEr [70]. Unlike the above metrics, CHRF [71] is an metric to evaluate the quality of generated sentences at the character level. CHRF++ [72] even integrates character level and word level, and takes the average of the two as the evaluation score. On the contrary, TTR (type-token ratio) and Dist-n [73] are the diversity measure used to evaluate the lexical richness of generated text. Specifically, Dist-n is the number of distinct n-grams divided by total number of generated words.

(2) Semantic Equivalence. More recently, metrics based on the similarity of sentence embeddings have shown improved correlations with human judgments at the sentence level. These metrics, including BERTScore [74], BLEURT [75], MoverScore [76], and FrugalScore [77], gain semantic equivalence between the generated text and the reference text by using the pre-trained language models. Besides, FINE and ROUGH [78] also use a pre-trained neural network model based on natural language inference (NLI) to assess the semantic accuracy of D2T, especially for identifying incorrect outputs.

(3) Faithfulness. In addition to measuring the semantic consistency, the faithfulness of the generated text is also a very important target. The unfaithful generation usually contains hallucinated content which can not be aligned to any input structured data, especially in table-to-text generation. Thus, PARENT [79] and PARENT-T [80] are proposed to evaluate the Faithfulness in table-to-text generation. RG, CS and CO [27] are three extraction evaluation indicators, which are usually used to evaluate the comprehensive effect (Relation Generation, Content Selection, and Content Ordering) of table-to-text generation models in experiments using RotoWire dataset. Similar to RG, ESA_{C^n} [81] can estimate the percentage of RDF entity collections that can find a corresponding mention in the generated text. Based on the alignment between source table records and recognized entities in the generated text, P_{cover} and R_{hallu} [82] can calculate the table record coverage and the ratio of hallucinated entities in generated text.

TABLE 2
List of current popular automatic evaluation metrics for D2T task. G: generated text. R: reference text. SenEmb: sentence embedding.

Category	Description	Metrics	Year
Lexical Similarity	Overlap of n-gram in (G, R) pair	BLEU	2002
		ROUGE	2004
		METEOR	2007
		NIST	2002
		CIDEr	2015
		CHRF	2015
		CHRF++	2017
	Lexical richness of G	TTR	2014
		Dist-n	2016
Semantic Equivalence	Similarity of SenEmb in (G, R)	BERTScore	2020
		BLEURT	2020
		MoverScore	2019
		FrugalScore	2022
	Natural language inference	FINE, ROUGH	2020
Faithfulness	Faithfulness of G to the input	PARENT	2019
		PARENT-T	2020
		RG, CS, CO	2017
		ESA_{C^n}	2021
		P_{cover} , R_{hallu}	2021

2.3.2 Human Evaluation Metrics

For D2T, human evaluation is implemented to assess on three metrics: *Fluency* (whether the generated text is fluent without grammatical error), *Faithfulness* (whether the generated text is faithful to input), and *Coherence* (whether the output is logically coherent and the order of expression is in line with human writing habits) [83].

2.4 Traditional Approaches to D2T

The traditional approaches of D2T follow the pipeline architecture for natural language generation. As shown in Figure 2, McKeown [1] proposed a classical pipelined text generation architecture that contained three stages: document planning, micro planning, and surface realization. Document planning and micro planning aim to solve the "What to say" problem, then surface realization aims to solve the "How to say" problem.

(1) *Document planning* is also known as text planning, discourse planning, or macro planning. It includes content determination and document structuring. Content determination aims to discover and determine the major topics the

text need to be covered. Document structuring needs to determine the overall structure of the text.

(2) *Micro planning* is also known as sentence planning which aims to convert a content plan into a sequence of sentences or phrases. It usually consists of three modules: aggregation, lexicalization, and referring expression generation. Aggregation is the process of grouping selected structured data together into sentences. Lexicalization is the process of deciding which specific words and phrases need to be chosen to express the domain concepts and relations between the structured data. Referring expression generation is the task of selecting suitable words and phrases to identify domain entities.

(3) *Surface realization* contains structure realization and linguistic realization. Structure realization aims to mark up the text's surface structure. Linguistic realization needs to smooth the text by insert function words, reorder word sequences, and select appropriate inflections and tenses of words.

Different from the three stages, Reiter and Dale [2] proposed another pipelined text generation architecture which consisted of six modules: content determination, discourse planning, sentence aggregation, lexicalization, referring expression generation, and linguistic realization. However, discourse planning is equivalent to document structuring in the document planning, and sentence aggregation is equivalent to aggregation in the micro planning.

The traditional modular approaches to D2T can be classified into two categories: (1) *template-based modular approaches* [84] and (2) *statistical modular approaches* [85]. The template-based modular approaches applied hand-engineered rules and templates to generate fixed format text. The statistical modular approaches are mostly based on probabilistic language generation process [3], probabilistic context-free grammar [4], and other probabilistic models. For example, Liang, Jordan, and Klein [63] proposed a probabilistic HSMM-based generative model which generates text from tabular data including a set of records in three stages: record choice, field choice, and word choice. It also used dynamic programming style decoding algorithm [7] when generating output text. Rather than breaking up the generation process into a sequence of local decisions, Konstas and Lapata [4] defined a probabilistic context-free grammar (PCFG) that globally describes the inherent structure of the input tabular data through a weighted hypergraph (or packed forest). Nevertheless, content planning (or document planning) was not executed in aforementioned probabilistic models. Therefore, Konstas and Lapata [86] represented content plans with grammar rules which are operated on the document level and are embedded on top of the original PCFG.

3 Neural Approaches to D2T

In recent years, neural end-to-end models for data-to-text generation have became dominant and achieved the SOTA results due to the rise of deep learning. Next, we first explain why neural end-to-end approaches for D2T. Then we survey neural end-to-end approaches to D2T in the light of different network frameworks, training and inference strategies. Finally, we discuss neural approaches with prior of dividing the task into different modules.

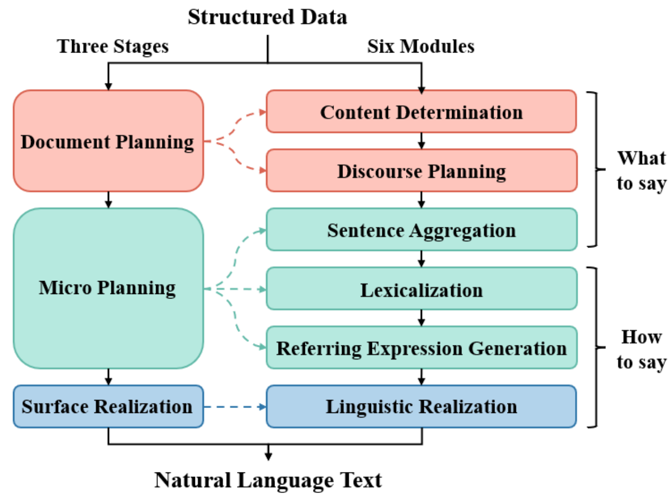

Fig. 2. The traditional pipeline architecture for data-to-text generation.

3.1 Why Neural End-to-End for D2T?

Although the traditional modular method has better interpretability, it still brings many problems. On the one hand, a multi-module cascading system will bring unavoidable error propagation problems; on the other hand, traditional modular method requires feature engineering involving more domain-specific handcrafted rules and templates, which are costly and not flexible enough.

In contrast, neural end-to-end approaches use the encoder-decoder architecture to effectively simulate the correlation between the output natural language text and the input structured data. This architecture unifies the functions of three stages (document planning, micro planning, and surface realization) into the decoder, avoiding the problem of error propagation. Moreover, using the data-driven neural network models can decrease the expense of creating domain-specific rules and templates by hand.

3.2 Frameworks of Neural End-to-End D2T

As shown in the Figure 3, the basic neural end-to-end D2T framework employed the attention-based encoder-decoder architecture, which was previously used for both Machine Translation [10] and Text Summarization [11]. In the basic neural end-to-end D2T framework, the general aim of D2T task is to find an optimal sequence Y conditioned on the input X via the teacher-forced maximum likelihood estimation (MLE), which can be formulated as:

$$Y = argmax_y \sum_{t=1}^{l} logP(y_t|Y_{<t}, X) \quad (1)$$

where l represents the number of tokens of the generated text Y, and $P(y_t|Y_{<t}, X)$ is the conditional probability of the next token y_t based on its previous tokens $Y_{<t}$ and the source structured data X.

Compared with introducing a coarse-to-fine aligner [87], the attention-based encoder-decoder architecture performed a better content selection [46]. Based on this basic framework, many studies have improved it on the following different components: (1) more powerful encoders are utilized to enhance structured data representation; (2) distinct

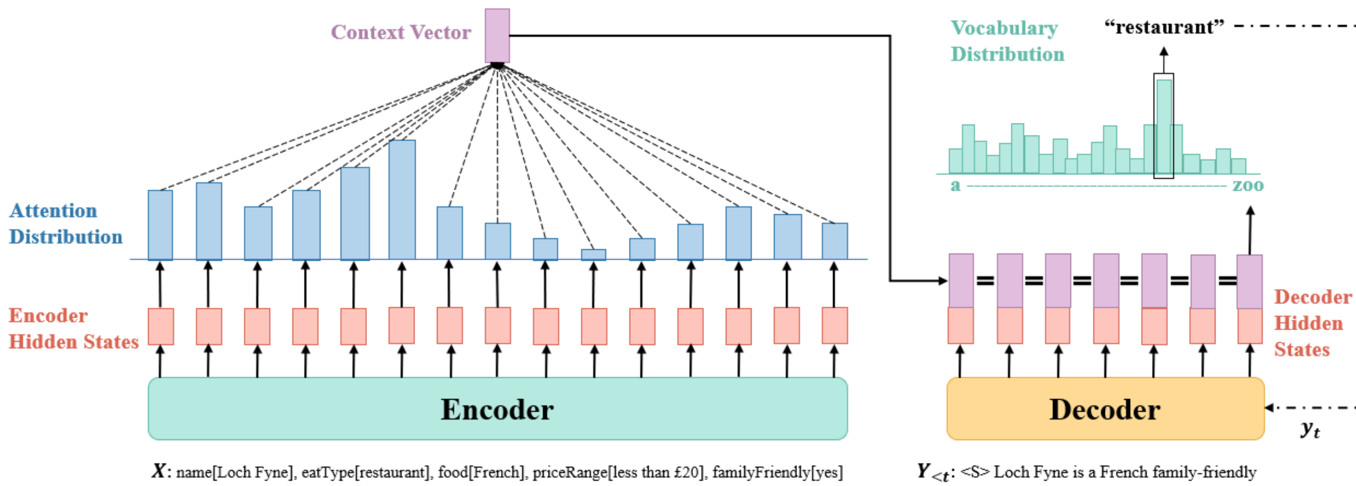

Fig. 3. The illustration of basic neural end-to-end data-to-text generation model which follows the attention-based encoder-decoder architecture.

decoding architectures are used to improve diversity, number reasoning, interpretability, and controllability. (3) copy and point mechanism is applied to deal with the out-of-vocabulary problem; (4) Transformer is utilized to better and faster model the relations between structured data and textual output; (5) to incorporate external knowledge, the multiple encoders and pre-training technique are applied.

3.2.1 More Powerful Encoder for D2T

Compared with convolutional neural network(CNN) and multi-layer perceptron (MLP), Recurrent Neural Networks (RNN) is more suitable for modeling natural language text with sequence structure.

(1) RNN-based Encoder. Widely-used variants of RNN in the D2T task include long-short-term memory (LSTM), GRU [88], and ARU [89]. However, the input structured data need to be linearized before the RNN-based encoder, which lost important structural information of the input. To address this issue, hierarchical attention mechanism [13], [14], [15] or dual attention mechanism [16], [17] was exploited in *Table-to-text generation* and *MR-to-text generation* to capture structural feature of records and fields. Most of the domain-specific tables are time series data (such as the RotoWire dataset), which means that the description of the current table may be affected by its historical data. Therefore, [90] simultaneously modeled row, column, and time dimension information by a hierarchical encoder to enhance the table representation. As to encode input graph, vanilla RNN requires specific modifications to capture the structural information of non-sequential data, such as GTR-LSTM triple encoder [91] and graph state LSTM [92].

(2) GNN-based Encoder. For the *Graph-to-text generation*, the Graph Neural Networks (GNN) was a better choice to encode the input graph than the LSTM. [93] replaced RNN with Graph Convolutional Networks (GCN) to explicitly encode the structural information of the input graph. [94] enhanced the representation of AMR graphs by explicitly encoding both top-down and bottom-up views of the input graph with different Graph Neural Networks, such as Gated Graph Neural Network [95], Graph Attention Network [96] and Graph Isomorphic Network [97]. [98] introduced the mix-order Graph Attention Networks as the graph encoder to model the relationships between indirectly connected nodes by integrating the higher-order neighborhood information. Lightweight Dynamic Graph Convolutional Networks (LDGCNs) [99] also capture richer non-local interactions by synthesizing higher order information through a dynamic fusion mechanism. DCGCN [100] utilized an artificial global node with direct edges to all other nodes to allow global message exchange, thereby neglecting graph topology as all nodes are directly connected. And the unified GAT structure [101] could encode an input graph combining both global and local node contexts, in order to learn better contextualized node embeddings.

(3) Multiple Encoders. The ability of a single encoder to model different features of the sequence is relatively limited. Thus, [102] created a neural ensemble natural language generator that accumulated the top 10 predicted utterances from three neural models with different encoder. Furthermore, [103] combined bi-GMP (a variant of GTR-LSTM triple encoder) and bi-GCN encoders in order to jointly learn the local and global structure information of the RDF triples. DUALENC [104] also adopted a dual encoding method (graph encoder and plan encoder) to narrow the structural gap between data encoder and text decoder.

3.2.2 Distinct Decoding Architectures for D2T

Decoding architectures can be divided into (i) Autoregressive and (ii) Non-autoregressive. Existing SOTA models in NLG tasks are mostly autoregressive where each generation step depends on the previously generated tokens.

(1) Autoregressive Decoding (AR) or Autoregressive Generation (AG) dynamically generates predictions in a recurrent manner. Predominant decoding strategies for generating text from autoregressive language models can be categorized into three classes: (1) maximization-based strategies, such as greedy search and beam search; (2) stochastic strategies, such as top-k sampling [105] and nucleus sampling [106]; (3) isotropic-based strategies, such as contrastive search [107]. Some works employed multiple decoders in a manner similar to that of multiple encoders in order to increase the description diversity of text generated. To be specific, [108] proposed a training method based on diverse ensembling to encourage models to learn latent distinct sentence templates by a shared encoder and K separate decoders. Multi-Branch Decoder (MBD) [109] also duplicated the decoder module into three distinct parallel modules associated with content, hallucination, and fluency. Neural Table Reasoning Generator (NTRG) [110] developed another RNN-based equation decoder to generate additional mathematical equations during decoding the target text, improving the number reasoning capability of neural D2T.

(2) Non-autoregressive Decoding (NAR) or Non-autoregressive Generation (NAG) that generates the entire or partial output sequences in parallel has faster inference speed than autoregressive decoding. [111] attempts to learn a non-autoregressive policy [112] that generates text by directly inserting or replacing spans of neighbor text at arbitrary positions within a partially constructed generation, which allows for more interpretable and controllable data-to-text generation.

3.2.3 Copy and Point Mechanism for D2T

The point mechanism proposed by Pointer Network [113] (Ptr-Net) facilitates copying words from the source text. Pointer-Generator Networks (PG-Net) [114] is a hybrid between basic neural end-to-end model and the Ptr-Net. It aims to deal with out-of-vocabulary (OOV) words through copying words from the source text via the pointer. The coverage mechanism was also used in Pointer-Generator Networks to discourage repetition when generating output words. Similar to the PG-Net, [115] and [116] also used the copy mechanism to obtain word from source data.

3.2.4 Whole Transformer for D2T

Transformer [117], released in 2017, follows the encoder-decoder architecture, using stacked multi-head scaled dot-product self-attention (MHSA) and point-wise, fully connected feed-forward networks (FFN) for both encoder and decoder. It allows for significantly more parallelization than the RNN. The MHSA mechanism can better capture the dependency in the input and enable the encoding to be less sensitive to any permutation noise [118]. Thus, Transformer can reach a new SOTA performance in NLP tasks.

Compared to LSTM-based model with hierarchical attention mechanism [13], [119] encoded the tabular data with the hierarchical transformer encoder and achieved better results in content selection. And [120] first applied the Transformer to the task of AMR-to-text generation and proposed a novel structure-aware self-attention approach to better model the relations between indirectly connected concepts in AMR graphs. In addition, Graph Transformer [121] used a pairwise interaction function to compute the semantic relations and used separate graph attentions on the incoming and outgoing neighbors, which help in enhanced capturing of the semantic information provided in the graph. Another version of the Graph Transformer [122] has a fully-connected view on arbitrary input AMR graphs, which allows direct communication between two distant nodes. GraphWriter [32] also uses a similar Graph Transformer to encode knowledge graphs in the AGENDA dataset. Compared to utilizing an artificial global node with direct edges to all other nodes in GNNs, the Graph Transformer can have easier access to global information in the input graph.

3.2.5 Incorporating external knowledge in D2T

The existing methods for incorporating external knowledge can be divided into two types: (1) explicit fusion through multiple encoders, typically, KBAtt [123] used the dual attention mechanism (table attention and knowledge base attention) on the table encoder and KB encoder to integrate external background knowledge for table-to-text generation; (2) implicit fusion through pre-training on external knowledge. The case in point is KGPT [35], which first pre-trained the novel Transformer-based architecture on the knowledge-grounded corpus KGTEXT and then fine-tuned it on downstream D2T datasets.

3.3 Training and Inference Strategies for Neural D2T

3.3.1 Multi-task Learning for D2T

Based on the assumption that related tasks have similar representation spaces, multi-task learning [124] leverages the same training data by introducing additional auxiliary tasks to enhance the representation learning capabilities of neural models. In D2T, the multi-task learning can be used to enhance the logical reasoning ability of model and obtain structural information of input data.

Logical Reasoning for Fidelity. The crucial problem for neural D2T is that, although being fluent and informative, models frequently generate faithless descriptions that are inconsistent with the structured data. A straightforward way to improve the fidelity of neural text generation is to separate symbolic operations from neural models [43]. While [125] utilized two auxiliary tasks, number ranking and importance ranking, to supervise the numerical table encoder to capture the numeric size relation and the importance relation between records in different dimensions. Content planning is another form of logical reasoning and can be strengthened by multi-task learning. [126] adapted the Transformer by modifying the input table representation (record embedding) and introducing an additional objective function for content selection modelling. [80] introduced two content-matching constraint losses, including a table-text disagreement constraint loss and a constrained content matching loss with optimal transport, in order to generate description texts faithful to tables.

Preserving Structural Information by Reconstruction. To learn better node representations of the input AMR graph, [127] proposed to optimize two simple but effective auxiliary reconstruction objectives: link prediction objective which requires predicting the semantic relationship between nodes, and distance prediction objective which requires predicting the distance between nodes. [128] also introduced two types of auto-encoding losses to structure-aware Transformer for preserving input information, individually focusing on different views of input graphs (linearized graphs and the set of triples). TableGPT exploited multi-task learning with two auxiliary tasks that preserved table's structural information by reconstructing the structure from GPT-2's representation and improved the fidelity by aligning the table and information in the generated text.

This article has been accepted for publication in IEEE Transactions on Knowledge and Data Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2023.3304385

TABLE 3
Summary of recent neural end-to-end approaches on D2T without using pre-training techniques. MTL: Multi-Task Learning. LM: Language Modeling. VI: Variational Inference. RL: Reinforcement Learning. B: BLEU. BTS: BERTScore. PAT: PARENT. CO: Content Ordering.

Work	Model			MTL	Strategy	Performance	Datasets
	Encoder	Attention	Decoder	(✔/✗)	(LM/VI/RL)	(B/BTS/PAT/...)	
Coarse2Fine [87]	LSTM	✗(Aligner)	LSTM	✗	LM	B:61.01% B:25.28%	WEATHERGOV RoboCup
Table NLM [46]	LSTM	✔	LSTM	✗	LM	B:34.7%	WikiBio
MELBOURNE [26]	LSTM	✔	LSTM	✗	LM	B:45.13%	WebNLG(A)
Condi-Copy [27]	MLP	✔	LSTM	✗	LM	B:14.49%, CO:8.68%	RotoWire
OpAtt [43]	GRU	✔	GRU	✗	LM	B:18.0% B:14.96%	ESPN RotoWire
Pointer+Pos [47]	Bi-GRU	✔(Slot-aware)	GRU	✗	LM	B:23.2%	Wikipedia
TGen [28]	LSTM	✔	LSTM	✗	LM	B:69.25%	E2E
Order planning [115]	LSTM	✔(Hybrid)	LSTM	✗	LM	B:43.91%	WikiBio
Two-level LSTM [14]	LSTM	✔(Hierarchical)	LSTM	✔	LM	B:44.14%	WikiBio
Richness RL [129]	LSTM	✔(Force)	LSTM	✗	RL	B:45.47%	WikiBio
FGT [130]	LSTM	✔	LSTM	✗	RL	B:15.73%, RG(P%:82.99)	RotoWire
ENT [13]	LSTM	✔(Hierarchical)	LSTM	✗	LM	B:16.12%, CO:20.17% B:11.51%, CO:24.51%	RotoWire MLB
KBAtt [123]	LSTM	✔(Dual)	LSTM	✗	LM	B:44.59%	WikiBio
Field-gating [16]	LSTM	✔(Dual)	LSTM	✗	LM	B:44.89%	WikiBio
Structural Enc. [17]	LSTM	✔(Dual)	LSTM	✗	LM	B:46.5%, PAT:54.0% B:67.5%, PAT:71.6%	WikiBio E2E
SeqGAN [51]	LSTM	✔(Dual)	LSTM	✗	RL	B:49.2%, PAT: 45.6% B:13.0%	WikiBio Wiki3C
Eff-Hie Enc. [90]	LSTM	✔(Hierarchical)	LSTM	✗	LM	B:16.24%, CO:20.70%	RotoWire
GTR-LSTM [91]	GTR-LSTM	✔	LSTM	✗	LM	B:58.6% B:34.1%	WebNLG(S) WebNLG(U)
Graph state LSTM [92]	Graph state LSTM	✔	LSTM	✗	LM	B:33.0%	AMR15
GCN Enc. [93]	GCN	✔	LSTM	✗	LM	B:53.5%	WebNLG
Dual Graph Enc. [94]	GNNs	✔	LSTM	✗	LM	B:24.32% B:27.87%	AMR15 AMR17
DCGCN [100]	DCGCN	✔	LSTM	✗	LM	B:30.4%, CHRF++:59.6%	AMR17
Mix-Order GAT [98]	4-MixGAT	✔	Trans.Dec	✗	LM	B:30.58% B:32.46%	AMR15 AMR17
LDGCNs [99]	LDGCNs	✔	LSTM	✗	LM	B:30.8%, CHRF++:61.8% B:33.6%, CHRF++:63.2% B:34.3%, CHRF++:63.7%	AMR15 AMR17 AMR20
CGE-LW [101]	CGE-LW	✔	Trans.Dec	✗	LM	B:18.01%, CHRF++:46.69% B:63.69%, CHRF++:76.66%	AGENDA WebNLG
GMP+GCN [103]	GMP+GCN	✔	LSTM	✗	LM	B:57.09% B:57.76%	WebNLG Enriched WebNLG
DUALENC [104]	DUALENC	✔	LSTM	✗	LM	B:63.45% B:36.73%	WebNLG(S) WebNLG(U)
Ensem. Enc. [102]	CNN+ Bi-LSTM	✔	LSTM	✗	LM	B:65.76%	E2E
Ensem. Dec. [108]	Bi-LSTM	✔	Multi-LSTM	✗	LM	B:74.3%	E2E
NTRG [110]	Bi-LSTM	✔	Multi-LSTM	✗	LM	B:18.03%, CO:22.46%	RotoWire
Hierarchical-k [119]	Trans.Enc(2)	✔(MHSA)	LSTM	✔	LM	B:17.5%, CO:18.90%	RotoWire
GatedGAT [125]	MLP+GatedGAT	✔(Dual)	LSTM	✔	LM	B:24.52%, CO:25.23% B:17.96%, CO:25.30%	RotoWire-FG RotoWire
DATA-TRANS [126]	Trans.Enc	✔(MHSA)	Trans.Dec	✔	LM	B:20.22%, CO:23.32%	RotoWire
IPOT [80]	Trans.Enc	✔(MHSA)	Trans.Dec	✔	LM	B:24.56%, PAT-T:56.1%	WikiPerson
Graph Trans. [121]	Trans.Enc	✔(MHGA)	Trans.Dec	✗	LM	B:25.9% B:29.3%, CHRF++:59.0%	AMR15 AMR17
Graph Trans. [122]	Bi-GRU	✔(Global)	Trans.Dec	✗	LM	B:27.4%, CHRF++:56.4% B:29.8%, CHRF++:59.4%	AMR15 AMR17
SA Trans. [120]	Trans.Enc	✔(SA)	Trans.Dec	✗	LM	B:29.66%, CHRF++:63.00% B:31.82%, CHRF++:64.05%	AMR15 AMR17
GR Trans. [127]	Trans.Enc	✔	Trans.Dec	✔	LM	B:32.1%, CHRF++:64.0% B:33.9%, CHRF++:65.8%	AMR15 AMR17
Multiview-G2S [128]	Trans.Enc	✔(SA)	Trans.Dec	✔	LM	B:34.21% B:62.89%	AMR17 WebNLG
Conf-PtGen [131]	Trans.Enc	✔	LSTM	✗	VI	B:65.58% B:44.21%, PAT(F1):54.35%	WebNLG WikiBio
Trans.+RL [132]	LSTM	✔	LSTM	✗	LM+RL(self-critical)	B:63.2%, PAT(F1):71.27% B:44.17%, PAT(F1):56.72%	WebNLG WikiBio
Inverse RL [133]	Bi-GRU	✔	GRU	✗	Inverse RL	B:28.42%	Wikipedia
SeqPlan [134]	LSTM	✔	LSTM	✗	VI (Interleaved)	B:16.26%, CO: 16.7% B:14.29%, CO: 22.7%	RotoWire MLB

3.3.2 Variational Inference for D2T

From MLE to Inverse KL Divergence. Training neural D2T models through maximum likelihood estimation (MLE) is the most widely used method. The objective of MLE is equivalent to minimizing the cross entropy between the real data distribution and the estimated probability distribution. While [135] regards the maximum likelihood estimation as minimizing the Kullback-Leibler (KL) divergence and then approximately optimizes the inverse Kullback-Leibler divergence between the distributions of the real and generated sentences during training the KG-to-Text generation model. Due to the nature that inverse KL imposes large penalty on fake-looking samples, this method can significantly reduce the probability of generating low-quality sentences.

What is VAE? Variational Auto-Encoder [136] is a type of likelihood-based generative model via variational inference to approximate the posterior of the model by maximizing the evidence lower bound (ELBO) which need to optimize the Kullback-Leibler divergence. When VAE is applied to sequence-to-sequence generation where the input and the output are denoted by X and Y respectively, its optimization objective is defined as follows:

$$\begin{aligned}\log P_\theta(Y|X) \geq E_{z\sim q_\phi(z|X,Y)}[\log P_\theta(Y|X,z)] \\ -KL(q_\phi(z|X,Y) \parallel P(z|X))\end{aligned} \tag{2}$$

where $P_\theta(Y|X,z)$ denotes the decoder with parameters θ, and $q_\phi(z|X,Y)$ is obtained by an encoder with parameters ϕ, and $P(z|X)$ is a prior distribution (e.g. the Gaussian distribution).

For logical table-to-text generation, the problem is that the causal relationship between the table and the text is more difficult to capture than the surface-level spurious correlations. DCVED [137] proposed a de-confounded variational encoder-decoder in view of causal intervention, utilizing variational inference to generate multiple candidates that finally selected by a table-text selector based on surface-level consistency and the logical fidelity. To address the issue of hallucination in D2T, [131] employed the Variational Bayes scheme to train the model so that it could generate confident text by learning from the sampled sub-sequences containing only parts of the target that were faithful to the source. Besides, [134] inferred latent plans sequentially with a structured variational inference model, while interleaving the steps of planning and generation for long-form text.

3.3.3 Reinforcement Learning for D2T

What is GAN? The generative adversarial network (GAN) [138] is a framework that adopts an adversarial training process to estimate generative models. GAN contains a generative model (G_ϕ) that captures the data distribution and tries to produce fake samples, and a discriminative model (D_θ) that attempts to determine whether the samples come from the model distribution or data distribution. The training objective with value function can be formulated as:

$$\begin{aligned}\min_{G_\phi}\max_{D_\theta} V\left(G_\phi, D_\theta\right) = E_{X\sim P_{real}}[logD_\theta(X)] \\ +E_{Z\sim P_{G_\phi}}[log(1-D_\theta(Z))]\end{aligned} \tag{3}$$

where X are the real samples that obey the distribution P_{real} , and Z are the generated samples which obey the distribution P_{G_ϕ}.

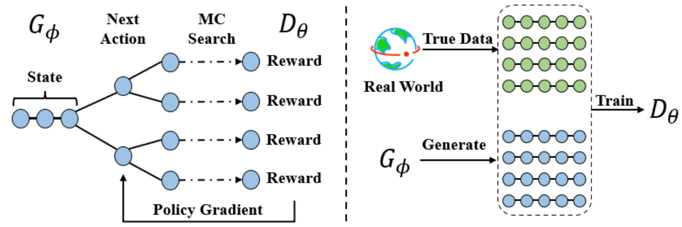

Fig. 4. The illustration of SeqGAN. Left: G_ϕ is trained by policy gradient where the final reward evaluated via Monte Carlo search is provided by D_θ. Right: D_θ is trained over the true data and the generated data.

GAN avoids the exposure bias problem caused by optimizing autoregressive model with MLE, and optimizes the Jensen-Shannon (JS) divergence between the real data distribution and the generative model distribution, which will have a reasonable penalty for generating false samples.

$$JS(P \parallel G) = \frac{1}{2}KL(P \parallel M) + \frac{1}{2}KL(G \parallel M) \tag{4}$$

where $M = \frac{1}{2}(P+G)$ is the average of two distributions.

However, due to the existence of non-derivable sampling operation, the traditional gradient optimization method can not be directly used to optimize GAN during training. One way to solve this problem is to transform it into a reinforcement learning problem that does not require derivability.

Train GAN using Reinforcement Learning (RL). The SeqGAN [139] considers the sequence generation procedure as a sequential decision making process. As illustrated in Figure 4, the generative model (G_ϕ) is treated as an agent (policy) of reinforcement learning and is trained via policy gradient; the state is the generated tokens so far and the action is the next token to be generated. To give the reward, SeqGAN employs a discriminative model (D_θ)to evaluate the sequence using Monte Carlo (MC) search to approximate the state-action value and feedbacks the evaluation to guide the learning of the generative model. For generating high-quality language descriptions, RankGAN [140] employs a ranker as the discriminative model to learn the relative ranking information between the machine-written and the human-written sentences. MaskGAN [141] even converts the text generation task into a fill-in-the-blank task that fills in missing text conditioned on the surrounding context.

In D2T, reinforcement learning can also improve the factual consistency and information richness of generated text. [51] adopted the SeqGAN to the table-to-text generation task and added a reward from the PARENT metric to represent the information fidelity to the input table. [129] took advantage of the reinforcement learning framework to encourage the LSTM-based MR-to-text generator to cover infrequent and rarely mentioned attributes, by maximizing a mix reward which contains both the BLEU score and the information richness of the generated description towards the source attribute-value pairs. [130] also designed a non-differentiable consistency verification signal which optimized via reinforcement learning in order to inspect fact discrepancy between generated texts and their corresponding input data. [142] proposed a new reward, Clinical Reconstruction Score (CRS), to quantify the factual correctness of reports with a BERT-based reconstructor, and designed the overall reward as a combination of ROUGE-L score and CRS. PARENTing [132] proposed a training protocol with a mixed objective function combining the standard maximum-likelihood loss with a custom reinforcement loss, which was optimized via self-critical policy gradient based on the PARENT score. While [133] posed the table-to-text generation task as Inverse Reinforcement Learning (IRL) problem and designed a set of intuitive and interpretable reward components that were linearly combined to get the reward function of Maximum Entropy IRL framework on the basis of [47].

3.3.4 *Pre-training for D2T*

Why Pre-training? In NLP, pre-trained models (PTMs) on large unannotated corpora have been proved to be beneficial for the downstream NLP tasks [159]. The advantages of pre-training technique can be summarized as follows: (1) Pre-training on the huge unannotated text corpora can learn universal language representations and help with the downstream tasks; (2) Pre-training provides a better model initialization, which usually leads to a better generalization performance and speeds up convergence on the downstream task; (3) Pre-training can be regarded as a kind of regularization to avoid overfitting on small data.

According to the architectures of backbone network, pre-trained language models (PLMs) can be divided into four categories: (1) LSTM as backbone network, including ELMo [160] , LM-LSTM [161], Shared LSTM [162], and CoVe [163]; (2) Transformer encoder (Trans.Enc) as backbone network, including BERT [164], XLNet [165], SpanBERT, RoBERTa [166], UniLM [167] and UniLM2 [168]; (3) Transformer decoder (Trans.Dec) as backbone network, including GPT [169], GPT-2 [170], and GPT-3 [171]; (4) full standard Transformer architecture as backbone network, including BART [172], T5 [173] and the multi-lingual version (mBART [174], mT5 [175], MASS [176], XNLG [177]). “Trans.Enc” and “Trans.Dec” mean the encoder and decoder part of the standard Transformer architecture, respectively. Their difference is that the decoder part uses masked self-attention with a triangular matrix to prevent tokens from attending their future (right) positions. Therefore, “Transformer decoder” is naturally suitable for text generation tasks. Denoising autoencoding as pre-training task used by BART can make the encoder robust to noisy input [178].

At present, the pre-trained language models commonly used in the D2T task involve: GPT-2, GPT-3, KGPT, T5, BART and so on. However, there is a significant gap in the adaptation of the universal pre-trained language models to specific downstream tasks. In D2T, this gap is mainly reflected in the difference between the topological information contained in structured data itself and the sequential text [179]. As shown in Figure 5, the two basic paradigms for utilizing the pre-trained language models for D2T are *Fine-tuning* and *Prompting*, which both aim at adapting the PLMs to D2T, thereby narrowing the gap above.

(1) ***Pre-training and Fine-tuning.***

Fine-tuning GPT-based model. In 2020, [143] explored the possibility of directly fine-tuning GPT-2 to predict target text on a sequential representation of AMR graphs. It outperformed the previous SOTA on automatic evaluation

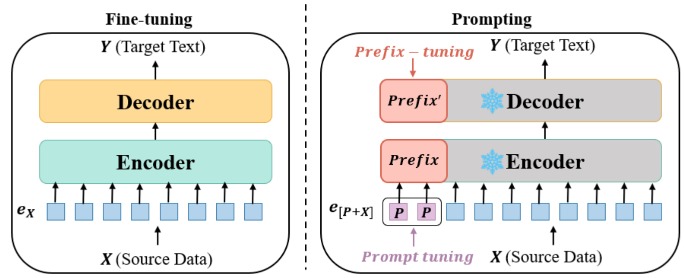

Fig. 5. The frameworks of two main paradigms to utilize the pre-trained models for D2T task. Left: Fine-tuning can update all parameters of the PTMs. Right: Prompting freezes the parameters of the PTMs (gray) and only updates small additional parameters for few-shot learning. It consists of prefix-tuning that prepends prefixes (red) at every transformer layer and prompt tuning that prepends the embeddings of prompt tokens (purple) to the embedded input.

metrics and human evaluation experiments for AMR-to-text generation. While [144] used the pre-trained language model GPT-2 as the generator, serving as the “innate language skill”, in the few-shot D2T model based on [16]. Similar to [144], [55] also applied GPT-2 as the generator to decode description from a table by fine-tuning GPT-2 on LogicNLG dataset. Instead of encoding the input table by additional table encoder, TableGPT [145] only employed GPT-2 as the whole model for few-shot Table-to-text generation. Structured table was reasonably transformed into text sequence by utilizing a table transformation module. After fine-tuning GPT-2 using the D2T task’s training data, [146] introduced a specialized semantic fidelity classifier (SFC) to assess how accurately the generated text reflects the input data. Besides, data augmentation is another frequently used method to alleviate the problem of insufficient training samples [180].

Fine-tuning T5/BART-based model. Directly adopting a pre-trained model with the encoder-decoder architecture, such as T5 and BART, for D2T is also an option [149] [150]. For instance, [151] fine-tuned T5 on D2T datasets by representing the structured data as a flat string (linearization), leading to state-of-the-art results on diverse benchmarks spanning MultiWOZ, ToTTo, and WebNLG. However, this method failed to fully utilize the structure information of the input data. To better capture the structure and interdependence of facts in the KG, [153] leveraged the power of T5 with two additional position embeddings, including triple role embeddings and tree-level embeddings. Instead of directly fine-tuning the PTMs on the training set, they first fine-tuned T5 on a noisy, but larger corpus crawled from Wikipedia before fine-tuning it on the WebNLG dataset. Similarly, STTP [154] further pre-trained BART on the preprocessed WDCWebTable corpus by three self-supervised tasks including masked table language modeling, adjacent cell prediction, and context reconstruction. [152] even investigated the robustness to permutation of graph linearization in T5-based graph-to-text generation models, and found that: 1) models trained on the fixed, canonical linearizations failed to generalize to meaning-preserving alternatives; 2) graph denoising objectives under the framework of multi-task learning would drive substantial improvements. [155] also proposed a relation-biased breadth-first search (BFS)

TABLE 4
Summary of recent neural end-to-end approaches on D2T with pre-training techniques. MTL: Multi-Task Learning. LM: Language Modeling. VI: Variational Inference. RL: Reinforcement Learning. B: BLEU. PAT: PARENT. PPL: Perplexity. BRT: BLEURT. FT: Fine-tuning. PT: Prefix-tuning.

Work	PTMs	Backbone	Paradigm	Year	Multi-task (✔/✗)	Strategy (LM/VI)	Performance (B/BTS/PAT/...)	Datasets
GPT-2 [55]	GPT-2	Trans.Dec	FT	2020	✗	LM	B:49.6%, PPL:6.8%	LogicNLG
DCVED [137]	GPT-2	Trans.Dec	FT	2021	✗	VI	B:48.9%, NLI-Acc: 73.8% B:49.5%, NLI-Acc: 76.9%	Logic2Text LogicNLG
GPT-2 [143]	GPT-2	Trans.Dec	FT	2020	✗	LM	B:33.02%, BTS(F1):94.63%	AMR17
LM+switch [144]	GPT-2	Trans.Dec	FT	2020	✗	LM	B-4(500 e.g.):41.7% B-4(500 e.g.):40.3% B-4(500 e.g.):42.2%	WikiBio Wikipedia(Books) Wikipedia(Songs)
TableGPT [145]	GPT-2	Trans.Dec	FT	2020	✔	LM	B-4(500 e.g.):45.6% B-4(500 e.g.):42.2% B-4(500 e.g.):42.3%	WikiBio Wikipedia(Books) Wikipedia(Songs)
DATATUNER [146]	GPT-2	Trans.Dec	FT	2020	✗	LM	B:52.9%, CIDEr:3.7 B:37.7%, CIDEr:3.9 B:43.6%, CIDEr:2.0 B:53.6%, CIDEr:2.7	WebNLG AMR17 Cleaned E2E ViGGO
KGPT [35]	KGPT	Trans.Dec	FT	2020	✗	LM	B:64.11% B:68.05% B:45.06%	WebNLG E2E WikiBio
AMG [147]	UniLM	Trans.Enc	FT	2020	✗	LM	B-4:49.02%, PAT:51.86%, PAT-T:44.7% B-4:43.88%, PAT:48.59%, PAT-T:42.69% B-4:45.09%, PAT:46.9%, PAT-T:37.36%	WikiBio Wikipedia(Books) Wikipedia(Songs)
NMT [148]	MASS-like	Transformer	FT	2020	✗	LM	B:26.3%, SER:1.9	Czech D2T
T5+STA [149]	T5	Transformer	FT	2020	✗	LM	B:59.70%, CHRF++:75.40% B:49.72%, BRT:0.6424 B:25.66%, CHRF++:63.7%, BRT:-0.089	WebNLG AMR17 AGENDA
CycleGT [150]	T5	Transformer	FT	2020	✗	LM	B:55.5%, CIDEr:3.81 B:44.56%, BRT:0.54 B:41.59%, CIDEr:3.57	WebNLG WebNLG+ GenWiki
T5-Large [151]	T5	Transformer	FT	2020	✗	LM	B:64.7% B:49.5%, PAT:58.4% B:35.1%	WebNLG ToTTo MultiWOZ
Scaffolding [152]	T5	Transformer	FT	2021	✔	LM	B:45.14%, BTS:76.54%	AMR17
Two-step FT [153]	T5	Transformer	FT Twice	2021	✗	LM	B:66.07%, BTS:96.21%, PAT:71.6%	WebNLG
STTP [154]	BART	Transformer	FT	2021	✗	LM	B:64.92% B:82.63%	WebNLG WEATHERGOV
RBFS [155]	BART	Transformer	FT	2021	✔	LM	B-4:61.88%, CIDEr:6.03 B-4:25.15%, CIDEr:3.23 B-4:48.46%, CIDEr:5.19	WebNLG AGENDA GenWiki
Prefix-tuning [156]	GPT-2	Trans.Dec	PT	2021	✗	LM	B:63.4%, TER:0.34 B:70.3%, CIDEr:2.47 B:46.7%, BTS:94%, BRT:0.40	WebNLG E2E DART
Control Prefixes [157]	T5	Transformer	PT	2021	✗	LM	B:67.32%, TER:0.3096 B:55.41%, TER:0.392, BRT:0.63 B:44.15%, CIDEr:2.04 B:53.6%, TER:0.4275, BTS:95%	WebNLG WebNLG+ Cleaned E2E DART
PCG [158]	BART	Transformer	PT	2022	✗	LM	B(500 e.g.):49.4%, PAT(500 e.g.):51.8% B(500 e.g.):45.6%, PAT(500 e.g.):49.3% B(500 e.g.):44.5%, PAT(500 e.g.):46.0%	WikiBio Wikipedia(Books) Wikipedia(Songs)

strategy for KG linearization in few-shot KG-to-text generation with BART-based model, which adopted multi-task learning with an auxiliary KG reconstruction task. For low-resource language scenarios, [148] pre-trained a 6-layers full Transformer model via machine translation task on a parallel corpus, previous to fine-tuning the model on the Czech D2T dataset [58].

(2) ***Pre-training and Prompting.*** As the parameter size of the pre-trained model continues to increase, so does the cost of fine-tuning the pre-trained model, due to this way modifies all parameters of the pre-trained model. While prompt learning methods [181] , instigated by the arrival of GPT-3, adapt pre-trained language models to downstream applications by using the task-specific prompt together with the input. Prompt tuning [182] uses a single prompt representation that is prepended to the embedded input. It only requires storing a small task-specific prompt for each task, and enables mixed-task inference using the original pre-trained model. For text generation tasks, prefix-tuning [156], [183] , which keeps language model parameters frozen and instead optimizes a sequence of continuous task-specific prompt vectors (prefix) that prepended at every transformer layer in PTMs, is more effective than prompt-embedding tuning. Control Prefixes [157] extends this method by combining prefix-tuning with controlled text generation, empowering the model to have finer-grained control during text generation. It incorporates attribute-level learnable representations into different layers of a pre-trained transformer, allowing for the generated text to be guided in a particular direction. Impressively, large language models which contain hundreds of billions (or more) of parameters can solve few-shot D2T through in-context learning [184].

3.4 Neural Modular Approaches to D2T

Although the data-driven end-to-end approaches are quite straightforward, avoiding the hassle of manually formulating templates and grammar rules, this method has two main drawbacks: (1) lack of explicit utilization of linguistic knowledge; (2) lack of effective means to control the quality of the generated text. By comparing pipeline and end-to-end approaches, [185] discovered that having explicit intermediate steps during generation resulted in better texts. [186] also demonstrated that properly aligning input sequences during training leads to highly controllable generation. It proves that introducing a content planning stage before text generation can effectively enhance controllability and faithfulness. Therefore, the two-stage method for neural D2T has been paid more and more attention by researchers. And neural template-based approaches also have been explored for interpretable and controllable D2T. Table 5 summarizes recent neural modular approaches on D2T.

3.4.1 Two-Stage: Content Planning and Text Generation

Relationship to traditional pipeline architecture. *Content Planning* aims to perform content selection and content ordering on the input structured data, outputting a content plan. This stage is equivalent to executing three modules in traditional pipeline architecture: content determination, discourse planning, and sentence aggregation. *Text Generation* aims to convert content plan into natural language text that describes input accurately and fluently. Essentially, it is equivalent to executing the other three modules in traditional pipeline architecture, including lexicalization, referring expression generation, and linguistic realization.

Neural content planning (NCP) [18] is a typical two-stage D2T model. It first generated a content plan, highlighting which information should be mentioned and in which order; and then the resultant content plan as input to the text generation stage. Dynamic content planning (NDP) [194] also dynamically selected the appropriate information from the input data during text generation. Analogously, two-stage method was used in the KG-to-text generation [19], [20]. The content plan usually consists of a sequence of facts in table [188], [189], or a sequence of predicates/relations in RDF triples, or a sequence of keys in MRs, following the order in references. Thus, planning-based hierarchical variational model (PHVM) [21] was conducive to long text generation by decomposing it into dependent shorter sentences generation sub-task, treated as macro content planning [195] at the paragraph level.

Although the content plan guides the text generation in the second stage, it has the potential to cause false cascades, resulting in a decrease in the quality of the generated text. Thus, it is extremely important to ensure the correctness of content plan in the first stage. To this end, [192] made two improvements on the basis of the NCP model: (1) designed contextual numeric value representations obtained through a pre-trained ranking task; (2) designed integrated rewards to verify content planning results. PlanGen [196] distinctively formulated the intermediate content plan as an ordered list of tokens, and regarded content planning as a sequence labeling task. Based on a simple but homologous content planner, PCG [158] adopted the prompt tuning in the second stage to control the factual contents and word order of the generated text via two kinds of prefixes, which were prepended to the input of BART encoder.

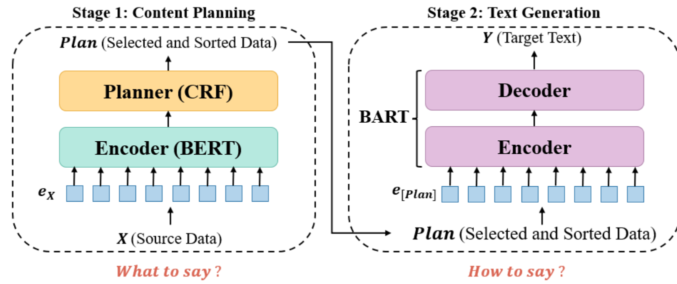

Fig. 6. The illustration of PlanGen model, which is a typical D2T model adopting two-stage method with PTMs. Left: stage 1 (content planning) aims to solve the "What to say" problem. Right: stage 2 (text generation) aims to sovle the "How to say" problem.

3.4.2 Neural Template-based Approaches to D2T

The neural template-based approaches [22], [187], [198], [199], in contrast to traditional approaches that manually constructed templates, first utilize the template generator to generate latent and discrete templates with data slots to be filled, after which it replicate the input's factual information to fill the slots. For example, [48] used the neural TextGen module to convert the modified canonical triple into a simple sentence-like template which would finally be replaced with the original entities to produce a simple sentence. Variational template machine (VTM) [23] explored the difficulty of automatically learning reusable templates from paired and non-paired data by means of variational inference. Besides, Anchor-to-Prototype [191] and Prototype-to-Generate [24] retrieved and selected a collection of prototype descriptions from the training data in order to guide the generation process. However, other research works [25], [193] used some simple handcrafted templates to transform each of the input triples into a single sentence, and then performed several simple modules to generate the final description, opening up the possibility for zero-shot domain adaptation. And [200] explicitly inserts the simple designed phrases corresponding to missing slots into the generated text to improve semantic coverage.

4 Applications of D2T

4.1 Potential applications of D2T in various fields

Due to the widespread presence of structured data in the real world, D2T models have many potential applications in various fields. For example, in the medical field, D2T can automatically generate the packaging leaflets of medicines [44] or medical examination reports [142]; in the field of journalism, D2T can be utilized to automatically generate biographies [26], [30], [31], weather forecasts [62], [63] and sports event broadcasts [13], [27], [41], [43], [59], [64]; in the financial field, D2T can be used to generate stock market comments [201] and restaurant reviews [28], [57], [58]; in the field of scientific research, D2T can also automatically generate experimental analysis reports [45]. These demonstrate the significant practical application value of D2T research.

TABLE 5
Summary of recent neural modular approaches on D2T. MTL: Multi-Task Learning. B: BLEU. BTS: BERTScore. PAT: PARENT. BRT: BLEURT.

Work	Backbone	Pre-trained	Paradigm	Two-stage	Template	MTL	Year	Performance	Datasets
HSMM [22]	LSTM	✗	✗	✗	✔	✗	2018	B:59.8%, NIST:7.56 B:34.8%, NIST:7.59	E2E WikiBio
DCM [187]	LSTM	✗	✗	✗	✔	✗	2018	B:16.19%, CO:16.34%	RotoWire
NCP [18]	LSTM	✗	✗	✔	✗	✗	2019	B:16.50%, CO:18.58%	RotoWire
3-stages [48]	LSTM	✗	✗	✗	✔	✗	2019	B:33.3%	WikiTablePara
BestPlan [19]	LSTM	✗	✗	✔	✗	✗	2019	B:47.4%, CIDEr:2.692	WebNLG
PIVOT [188]	LSTM	✗	✗	✔	✗	✗	2019	B:27.34%, NIST:6.8763	WikiBio
CSP+TG [189]	Trans.	✗	✗	✔	✗	✔	2019	B:15.17%, CO:19.26%	RotoWire-Modified
Segment. [190]	LSTM	✗	✗	✔	✗	✗	2020	B:65.1%, Dist-3:911 B:46.1%, Dist-3:149	E2E WebNLG
Anc2Pro [191]	Trans.	✗	✗	✗	✔	✔	2020	B:49.9%	WebNLG
DUV [192]	LSTM	✗	✗	✔	✗	✔	2020	B:15.92%, CO:23.32% B:9.51%, CO:27.78%	RotoWire-Modified MLB
ITE [193]	T5	✔	Fine-tuning	✗	✔	✗	2020	B:57.1%	WebNLG
NDP [194]	LSTM	✗	✗	✔	✗	✗	2021	B:16.67%, CO:20.67%	RotoWire
Macro [195]	LSTM	✗	✗	✔	✗	✗	2021	B:15.46%, CO:17.7% B:12.62%, CO:21.8%	RotoWire MLB
PlanGen [196]	BERT/BART	✔	Fine-tuning	✔	✗	✗	2021	B:65.42% B:49.2%, PAT:58.7%, BRT:0.249	WebNLG ToTTo
Aug-plan [82]	BART	✔	Fine-tuning	✔	✗	✗	2021	B:31.16%, PAT:56.75%	WikiPerson
SANA [197]	Trans.	✗	✗	✔	✗	✔	2021	B:54.51%, PAT:61.01% B:30.29%, PAT:68.28%	WikiBio WikiPerson
P2G [24]	T5	✔	Prefix-tuning	✗	✔	✔	2021	B(500 e.g.):50.1%	WikiBio
3-STAGE [25]	BART	✔	✗	✗	✔	✗	2022	B:43.94%(Zero-Shot) B:36.04%(Zero-Shot)	WebNLG E2E

4.2 The wider social impact of D2T

Even though the D2T models can be applied in various fields to help people analyze and illustrate structured data, there is still a crucial issue that needs to be addressed urgently, which is the hallucination problem. This problem can be further divided into two categories: (1) Intrinsic hallucination, which means that the generated text contains information that is unfaithful to the input; (2) Extrinsic hallucination, which means that the generated text contains non-factual information. Both intrinsic and extrinsic hallucinations can cause extremely serious consequences for the D2T models in practical applications, especially in the healthcare field. To mitigate the impact of hallucinations, SANA [197] introduced the non-autoregressive model into the two-stage method, resulting in fewer hallucinations. To eliminate the influence of hallucinations contained in training corpus, [82] incorporated the auxiliary entity information into the augmented plan and trained the plan-to-text generator with the few-shot high-quality augmented plan and text pairs. And [190] proposed to explicitly segment target text into fragment units and align them with their data correspondences, which reduced intrinsic hallucinations.

5 Challenges And Future Directions

5.1 Challenges

Although existing neural data-to-text generation models can generate fluent descriptions, there are still a number of issues that need to be further addressed. Current challenges mainly include the following five aspects:

(1) Weak Controllability. It is challenging to control the text generation process using neural end-to-end D2T models. Even though the two-stage method can somewhat increase the controllability of generated content, there is still a dearth of study on the style controllability of D2T.

(2) Low Coverage. Due to the weak controllability in the generation process, the text generated by the neural D2T model cannot fully describe the important information in the input. For instance, some slot information in MRs is omitted in the descriptions.

(3) Poor Faithfulness. Even if the generated text mentions all crucial information, it may contain content, called hallucinations, inconsistent with the facts. This issue is particularly evident in logical table-to-text generation which requires numerical reasoning and becomes a significant performance bottleneck.

(4) Inadequate Description Diversity. The same semantic meaning can be expressed through different descriptions. However, traditional automatic evaluation metrics mainly measure the word similarity between the generated text and the reference text, which limits the linguistic diversity [202].

(5) Loss of Structural Information. Linearization of structured data severely loses its structural information. Although the existing work can enhance the model's ability to capture the topological structure of the input data through multi-task learning [127], [128], [145] and adding extra embedding vectors [13], [14], [15], [16], [17], [104], [123], the model's ability to understand and restore the structure from linearized tables and graphs remains insufficient.

5.2 Future Directions

Based on the above-mentioned challenges, future research directions can be carried out from the following five points:

- Incorporating additional prior background knowledge [203] into the learning process is conducive to generate faithful and informative description via fact checking [204].
- Since D2T will be applied in different domains and scenarios, such as empathetic dialogue [205] in which

the style of generated text needs to be controlled based on the user's response, so we expect the D2T model to be able to describe the same content in different styles in a controlled manner [206]. Thus, style controllability of D2T will become an important research direction and promote the development of diversified descriptions.

- A better encoding technique is required to capture the structural information in the input data since there is still a topological structure gap between structured data and linearization.
- The existing pre-trained language models (PLMs) have the weakness of lacking the ability to perceive and reason about numerical values. Consequently, how to effectively enhance the numerical reasoning ability of the PLMs-based D2T model will be a critical research focus.
- It is also necessary to develop easy-to-use toolkits and sophisticated systems for D2T [207], [208], [209].

6 Conclusion

The goal of this survey is to review recent studies on neural D2T approaches in order to help new researchers in building a comprehensive understanding of this field. We include in this survey the background of the D2T research, a synopsis of traditional techniques, the state-of-the-art at the moment, challenges, and potential directions for future research. First, we introduce preliminaries such as the definition of D2T task, the three categories of D2T datasets that are currently accessible, evaluation metrics based on various measures, and traditional approaches to D2T. Next, we perform a taxonomy survey based on two axes: neural end-to-end D2T and neural modular D2T. We briefly illustrate the advantages of the neural end-to-end D2T and the composition of its basic framework, and further introduce the different deep learning technologies applied in D2T. In neural modular D2T, we summarize the existing work of integrating content planning and templates into the neural D2T, which aims to improve controllability and interpretability. We also discuss the potential applications in various fields and the adverse impacts of D2T. Finally, we present readers with the challenges in neural D2T and suggest some potential future directions. We hope research in this field can benefit from this survey.

References

[1] K. R. McKeown, *Text generation - using discourse strategies and focus constraints to generate natural language text*, ser. Studies in natural language processing. Cambridge University Press, 1992.

[2] E. Reiter and R. Dale, "Building applied natural language generation systems," *Nat. Lang. Eng.*, vol. 3, no. 1, pp. 57–87, 1997.

[3] G. Angeli, P. Liang, and D. Klein, "A simple domain-independent probabilistic approach to generation," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP 2010, 9-11 October 2010, MIT Stata Center, Massachusetts, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 502–512.

[4] I. Konstas and M. Lapata, "Unsupervised concept-to-text generation with hypergraphs," in *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 3-8, 2012, Montréal, Canada*, pp. 752–761.

[5] R. Barzilay and M. Lapata, "Collective content selection for concept-to-text generation," in *HLT/EMNLP 2005, Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 6-8 October 2005, Vancouver, British Columbia, Canada*, pp. 331–338.

[6] A. Belz, "Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models," *Nat. Lang. Eng.*, vol. 14, no. 4, pp. 431–455, 2008.

[7] J. Kim and R. J. Mooney, "Generative alignment and semantic parsing for learning from ambiguous supervision," in *COLING 2010, 23rd International Conference on Computational Linguistics, Posters Volume, 23-27 August 2010, Beijing, China*, pp. 543–551.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.

[9] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pp. 3104–3112.

[10] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

[11] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pp. 379–389.

[12] J. Li, W. Monroe, A. Ritter, D. Jurafsky, M. Galley, and J. Gao, "Deep reinforcement learning for dialogue generation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 1192–1202.

[13] R. Puduppully, L. Dong, and M. Lapata, "Data-to-text generation with entity modeling," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 2023–2035.

[14] T. Liu, F. Luo, Q. Xia, S. Ma, B. Chang, and Z. Sui, "Hierarchical encoder with auxiliary supervision for neural table-to-text generation: Learning better representation for tables," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 6786–6793.

[15] J. Cao, J. Gong, and P. Zhang, "Two-level model for table-to-text generation," in *Proceedings of the 2019 International Symposium on Signal Processing Systems*, 2019, pp. 121–124.

[16] T. Liu, K. Wang, L. Sha, B. Chang, and Z. Sui, "Table-to-text generation by structure-aware seq2seq learning," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 4881–4888.

[17] J. Mahapatra and U. Garain, "Exploring structural encoding for data-to-text generation," in *Proceedings of the 14th International Conference on Natural Language Generation, INLG 2021, Aberdeen, Scotland, UK, 20-24 September, 2021*, pp. 404–415.

[18] R. Puduppully, L. Dong, and M. Lapata, "Data-to-text generation with content selection and planning," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 6908–6915.

[19] A. Moryossef, Y. Goldberg, and I. Dagan, "Step-by-step: Separating planning from realization in neural data-to-text generation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 2267–2277.

[20] ——, "Improving quality and efficiency in plan-based neural data-to-text generation," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, pp. 377–382.

[21] Z. Shao, M. Huang, J. Wen, W. Xu, and X. Zhu, "Long and diverse text generation with planning-based hierarchical variational model," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 3255–3266.

[22] S. Wiseman, S. M. Shieber, and A. M. Rush, "Learning neural templates for text generation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 3174–3187.

[23] R. Ye, W. Shi, H. Zhou, Z. Wei, and L. Li, "Variational template machine for data-to-text generation," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

[24] Y. Su, Z. Meng, S. Baker, and N. Collier, "Few-shot table-to-text generation with prototype memory," in *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pp. 910–917.

[25] Z. Kasner and O. Dusek, "Neural pipeline for zero-shot data-to-text generation," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 3914–3932.

[26] C. Gardent, A. Shimorina, S. Narayan, and L. Perez-Beltrachini, "The webnlg challenge: Generating text from rdf data," in *Proceedings of the 10th International Conference on Natural Language Generation*, 2017, pp. 124–133.

[27] S. Wiseman, S. M. Shieber, and A. M. Rush, "Challenges in data-to-document generation," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pp. 2253–2263.

[28] J. Novikova, O. Dusek, and V. Rieser, "The E2E dataset: New challenges for end-to-end generation," in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, Saarbrücken, Germany, August 15-17, 2017*, pp. 201–206.

[29] M. Sharma, A. Gogineni, and N. Ramakrishnan, "Innovations in neural data-to-text generation," *CoRR*, vol. abs/2207.12571, 2022.

[30] T. C. Ferreira, D. Moussallem, E. Krahmer, and S. Wubben, "Enriching the webnlg corpus," in *Proceedings of the 11th International Conference on Natural Language Generation*, 2018, pp. 171–176.

[31] T. C. Ferreira, C. Gardent, N. Ilinykh, C. V. D. Lee, S. Mille, D. Moussallem, and A. Shimorina, "The 2020 bilingual, bidirectional webnlg+ shared task overview and evaluation results (webnlg+ 2020)," in *Natural Language Generation*, 2020.

[32] R. Koncel-Kedziorski, D. Bekal, Y. Luan, M. Lapata, and H. Hajishirzi, "Text generation from knowledge graphs with graph transformers," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1*, pp. 2284–2293.

[33] Z. Fu, B. Shi, W. Lam, L. Bing, and Z. Liu, "Partially-aligned data-to-text generation with distant supervision," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, November 16-20, 2020*, pp. 9183–9193.

[34] L. Nan, D. Radev, R. Zhang, A. Rau, and N. F. Rajani, "Dart: Open-domain structured data record to text generation," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021.

[35] W. Chen, Y. Su, X. Yan, and W. Y. Wang, "KGPT: knowledge-grounded pre-training for data-to-text generation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, November 16-20, 2020*, pp. 8635–8648.

[36] A. Colas, A. Sadeghian, Y. Wang, and D. Z. Wang, "Eventnarrative: A large-scale event-centric dataset for knowledge graph-to-text generation," 2021.

[37] Z. Jin, Q. Guo, X. Qiu, and Z. Zhang, "Genwiki: A dataset of 1.3 million content-sharing text and graphs for unsupervised graph-to-text generation," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 2398–2409.

[38] O. Agarwal, H. Ge, S. Shakeri, and R. Al-Rfou, "Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pp. 3554–3565.

[39] K. Knight, B. Badarau, L. Baranescu, C. Bonial, M. Bardocz, K. Griffitt, U. Hermjakob, D. Marcu, M. Palmer, and T. O'Gorman, "Abstract meaning representation (amr) annotation release 2.0."

[40] K. Knight, B. Badarau, L. Baranescu, C. Bonial, M. Bardocz, K. Griffitt, U. Hermjakob, D. Marcu, M. Palmer, T. O'Gorman *et al.*, "Abstract meaning representation (amr) annotation release 3.0," 2021.

[41] H. Iso, Y. Uehara, T. Ishigaki, H. Noji, E. Aramaki, I. Kobayashi, Y. Miyao, N. Okazaki, and H. Takamura, "Learning to select, track, and generate for data-to-text," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 2102–2113.

[42] H. Wang, "Revisiting challenges in data-to-text generation with fact grounding," *CoRR*, vol. abs/2001.03830, 2020.

[43] F. Nie, J. Wang, J.-G. Yao, R. Pan, and C.-Y. Lin, "Operation-guided neural networks for high fidelity data-to-text generation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Oct. 2018, pp. 3879–3889.

[44] R. Yermakov, N. Drago, and A. Ziletti, "Biomedical data-to-text generation via fine-tuning transformers," in *Proceedings of the 14th International Conference on Natural Language Generation, INLG 2021, Aberdeen, Scotland, UK, 20-24 September, 2021*, pp. 364–370.

[45] L. H. Suadaa, H. Kamigaito, K. Funakoshi, M. Okumura, and H. Takamura, "Towards table-to-text generation with numerical reasoning," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, Volume 1, Virtual Event, August 1-6, 2021*, pp. 1451–1465.

[46] R. Lebret, D. Grangier, and M. Auli, "Neural text generation from structured data with application to the biography domain," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 1203–1213.

[47] Q. Wang, X. Pan, L. Huang, B. Zhang, Z. Jiang, H. Ji, and K. Knight, "Describing a knowledge base," in *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pp. 10–21.

[48] A. Laha, P. Jain, A. Mishra, and K. Sankaranarayanan, "Scalable micro-planned generation of discourse from structured data," *Comput. Linguistics*, vol. 45, no. 4, pp. 737–763, 2019.

[49] M. Chen, S. Wiseman, and K. Gimpel, "Wikitablet: A large-scale data-to-text dataset for generating wikipedia article sections," in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, pp. 193–209.

[50] J. Bao, D. Tang, N. Duan, Z. Yan, Y. Lv, M. Zhou, and T. Zhao, "Table-to-text: Describing table region with natural language," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 5020–5027.

[51] J. Zhao, Z. Zhan, T. Li, R. Li, C. Hu, S. Wang, and Y. Zhang, "Generative adversarial network for table-to-text generation," *Neurocomputing*, vol. 452, pp. 28–36, 2021.

[52] A. P. Parikh, X. Wang, S. Gehrmann, M. Faruqui, B. Dhingra, D. Yang, and D. Das, "Totto: A controlled table-to-text generation dataset," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 1173–1186.

[53] W. Chen, H. Wang, J. Chen, Y. Zhang, H. Wang, S. Li, X. Zhou, and W. Y. Wang, "Tabfact: A large-scale dataset for table-based fact verification," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

[54] T. Li, L. Fang, J. Lou, and Z. Li, "TWT: table with written text for controlled data-to-text generation," in *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pp. 1244–1254.

[55] W. Chen, J. Chen, Y. Su, Z. Chen, and W. Y. Wang, "Logical natural language generation from open-domain tables," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 7929–7942.

[56] Z. Chen, W. Chen, H. Zha, X. Zhou, Y. Zhang, S. Sundaresan, and W. Y. Wang, "Logic2text: High-fidelity natural language generation from logical forms," in *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, pp. 2096–2111.

[57] O. Dusek, D. M. Howcroft, and V. Rieser, "Semantic noise matters for neural natural language generation," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, pp. 421–426.

[58] O. Dusek and F. Jurcícek, "Neural generation for czech: Data and baselines," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, pp. 563–574.

[59] J. Juraska, K. Bowden, and M. A. Walker, "Viggo: A video game corpus for data-to-text generation in open-domain conversation," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, pp. 164–172.

[60] M. Eric, L. Krishnan, F. Charette, and C. D. Manning, "Key-value retrieval networks for task-oriented dialogue," in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, Saarbrücken, Germany, August 15-17, 2017*, pp. 37–49.

[61] P. Budzianowski, T. Wen, B. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gasic, "Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 5016–5026.

[62] S. G. Sripada, J. Yu, I. P. Davy, and W. Oceanroutes, "Sumtime-meteo: Parallel corpus of naturally occurring forecast texts and weather data," 2008.

[63] P. Liang, M. I. Jordan, and D. Klein, "Learning semantic correspondences with less supervision," in *ACL 2009, Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2-7 August 2009, Singapore*, 2009, pp. 91–99.

[64] D. L. Chen and R. J. Mooney, "Learning to sportscast: a test of grounded language acquisition," in *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, ser. ACM International Conference Proceeding Series, vol. 307, pp. 128–135.

[65] C. van der Lee, C. Emmery, S. Wubben, and E. Krahmer, "The CACAPO dataset: A multilingual, multi-domain dataset for neural pipeline and end-to-end data-to-text generation," in *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pp. 68–79.

[66] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pp. 311–318.

[67] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[68] A. Lavie and A. Agarwal, "METEOR: an automatic metric for MT evaluation with high levels of correlation with human judgments," in *Proceedings of the Second Workshop on Statistical Machine Translation, WMT@ACL 2007, Prague, Czech Republic, June 23, 2007*, pp. 228–231.

[69] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *Proceedings of the second international conference on Human Language Technology Research*, 2002, pp. 138–145.

[70] R. Vedantam, C. L. Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pp. 4566–4575.

[71] M. Popovic, "chrf: character n-gram f-score for automatic MT evaluation," in *Proceedings of the Tenth Workshop on Statistical Machine Translation, WMT@EMNLP 2015, 17-18 September 2015, Lisbon, Portugal*, pp. 392–395.

[72] ——, "chrf++: words helping character n-grams," in *Proceedings of the Second Conference on Machine Translation, WMT 2017, Copenhagen, Denmark, September 7-8, 2017*, pp. 612–618.

[73] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," in *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pp. 110–119.

[74] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with BERT," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

[75] T. Sellam, D. Das, and A. P. Parikh, "BLEURT: learning robust metrics for text generation," *CoRR*, vol. abs/2004.04696, 2020.

[76] W. Zhao, M. Peyrard, F. Liu, Y. Gao, C. M. Meyer, and S. Eger, "Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance," *CoRR*, vol. abs/1909.02622, 2019.

[77] M. K. Eddine, G. Shang, A. J. Tixier, and M. Vazirgiannis, "Frugalscore: Learning cheaper, lighter and faster evaluation metrics for automatic text generation," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 1305–1318.

[78] O. Dusek and Z. Kasner, "Evaluating semantic accuracy of data-to-text generation with natural language inference," *CoRR*, vol. abs/2011.10819, 2020.

[79] B. Dhingra, M. Faruqui, A. P. Parikh, M. Chang, D. Das, and W. W. Cohen, "Handling divergent reference texts when evaluating table-to-text generation," *CoRR*, vol. abs/1906.01081, 2019.

[80] Z. Wang, X. Wang, B. An, D. Yu, and C. Chen, "Towards faithful neural table-to-text generation with content-matching constraints," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 1072–1086.

[81] J. Faille, A. Gatt, and C. Gardent, "Entity-based semantic adequacy for data-to-text generation," in *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pp. 1530–1540.

[82] T. Liu, X. Zheng, B. Chang, and Z. Sui, "Towards faithfulness in open domain table-to-text generation from an entity-centric view," in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 13 415–13 423.

[83] C. Dong, Y. Li, H. Gong, M. Chen, J. Li, Y. Shen, and M. Yang, "A survey of natural language generation," *CoRR*, vol. abs/2112.11739, 2021.

[84] K. Kukich, "Design of a knowledge-based report generator," in *21st Annual Meeting of the Association for Computational Linguistics*, Cambridge, Massachusetts, USA, Jun. 1983, pp. 145–150.

[85] I. Langkilde and K. Knight, "Generation that exploits corpus-based statistical knowledge," in *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, COLING-ACL '98, August 10-14, 1998, Université de Montréal, Montréal, Quebec, Canada. Proceedings of the Conference*, 1998, pp. 704–710.

[86] I. Konstas and M. Lapata, "Inducing document plans for concept-to-text generation," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1503–1514.

[87] H. Mei, M. Bansal, and M. R. Walter, "What to talk about and how? selective generation using lstms with coarse-to-fine alignment," in *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pp. 720–730.

[88] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1724–1734.

[89] H. Wang, W. Zhang, Y. Zhu, and Z. Bai, "Data-to-text generation with attention recurrent unit," in *International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019*, pp. 1–8.

[90] H. Gong, X. Feng, B. Qin, and T. Liu, "Table-to-text generation with effective hierarchical encoder on three dimensions (row, column and time)," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 3141–3150.

[91] B. D. Trisedya, J. Qi, R. Zhang, and W. Wang, "GTR-LSTM: A triple encoder for sentence generation from RDF data," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 1627–1637.

[92] L. Song, Y. Zhang, Z. Wang, and D. Gildea, "A graph-to-sequence model for amr-to-text generation," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 1616–1626.

[93] D. Marcheggiani and L. Perez-Beltrachini, "Deep graph convolutional encoders for structured data to text generation," in *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pp. 1–9.

[94] L. F. R. Ribeiro, C. Gardent, and I. Gurevych, "Enhancing amr-to-text generation with dual graph representations," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, November 3-7, 2019*, pp. 3181–3192.

[95] Y. Li, D. Tarlow, M. Brockschmidt, and R. S. Zemel, "Gated graph sequence neural networks," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

[96] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

[97] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

[98] Y. Zhao, L. Chen, Z. Chen, R. Cao, S. Zhu, and K. Yu, "Line graph enhanced amr-to-text generation with mix-order graph attention networks," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 732–741.

[99] Y. Zhang, Z. Guo, Z. Teng, W. Lu, S. B. Cohen, Z. Liu, and L. Bing, "Lightweight, dynamic graph convolutional networks for amr-to-text generation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 2162–2172.

[100] Z. Guo, Y. Zhang, Z. Teng, and W. Lu, "Densely connected graph convolutional networks for graph-to-sequence learning," *Trans. Assoc. Comput. Linguistics*, vol. 7, pp. 297–312, 2019.

[101] L. F. R. Ribeiro, Y. Zhang, C. Gardent, and I. Gurevych, "Modeling global and local node contexts for text generation from knowledge graphs," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 589–604, 2020.

[102] J. Juraska, P. Karagiannis, K. Bowden, and M. A. Walker, "A deep ensemble model with slot alignment for sequence-to-sequence natural language generation," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pp. 152–162.

[103] H. Gao, L. Wu, P. Hu, and F. Xu, "Rdf-to-text generation with graph-augmented structural neural encoders," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pp. 3030–3036.

[104] C. Zhao, M. A. Walker, and S. Chaturvedi, "Bridging the structural gap between encoding and decoding for data-to-text generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 2481–2491.

[105] A. Fan, M. Lewis, and Y. N. Dauphin, "Hierarchical neural story generation," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 889–898.

[106] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

[107] Y. Su and N. Collier, "Contrastive search is what you need for neural text generation," *CoRR*, vol. abs/2210.14140, 2022.

[108] S. Gehrmann, F. Z. Dai, H. Elder, and A. M. Rush, "End-to-end content and plan selection for data-to-text generation," in *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pp. 46–56.

[109] C. Rebuffel, M. Roberti, L. Soulier, G. Scoutheeten, R. Cancelliere, and P. Gallinari, "Controlling hallucinations at word level in data-to-text generation," *Data Min. Knowl. Discov.*, vol. 36, no. 1, pp. 318–354, 2022.

[110] X. Feng, H. Gong, Y. Chen, Y. Sun, B. Qin, W. Bi, X. Liu, and T. Liu, "Learning number reasoning for numerical table-to-text generation," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 8, pp. 2269–2280, 2021.

[111] S. Wiseman, A. Backurs, and K. Stratos, "Data-to-text generation by splicing together nearest neighbors," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 4283–4299.

[112] J. Gu, J. Bradbury, C. Xiong, V. O. K. Li, and R. Socher, "Non-autoregressive neural machine translation," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

[113] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 2692–2700.

[114] A. See, P. J. Liu, and 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, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pp. 1073–1083.

[115] L. Sha, L. Mou, T. Liu, P. Poupart, S. Li, B. Chang, and Z. Sui, "Order-planning neural text generation from structured data," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 5414–5421.

[116] N. Jiang, J. Chen, R. Zhou, C. Wu, H. Chen, J. Zheng, and T. Wan, "PAN: pipeline assisted neural networks model for data-to-text generation in social internet of things," *Inf. Sci.*, vol. 530, pp. 167–179, 2020.

[117] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 5998–6008.

[118] F. Nie, J. Wang, R. Pan, and C. Lin, "An encoder with non-sequential dependency for neural data-to-text generation," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, pp. 141–146.

[119] C. Rebuffel, L. Soulier, G. Scoutheeten, and P. Gallinari, "A hierarchical model for data-to-text generation," in *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part I*, ser. Lecture Notes in Computer Science, vol. 12035, pp. 65–80.

[120] J. Zhu, J. Li, M. Zhu, L. Qian, M. Zhang, and G. Zhou, "Modeling graph structure in transformer for better amr-to-text generation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 5458–5467.

[121] T. Wang, X. Wan, and H. Jin, "Amr-to-text generation with graph transformer," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 19–33, 2020.

[122] D. Cai and W. Lam, "Graph transformer for graph-to-sequence learning," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 7464–7471.

[123] S. Chen, J. Wang, X. Feng, F. Jiang, B. Qin, and C. Lin, "Enhancing neural data-to-text generation models with external background knowledge," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International*

Joint Conference on Natural Language Processing, Hong Kong, China, November 3-7, 2019, pp. 3020–3030.

[124] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.

[125] L. Li, C. Ma, Y. Yue, and D. Hu, "Improving encoder by auxiliary supervision tasks for table-to-text generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pp. 5979–5989.

[126] L. Gong, J. M. Crego, and J. Senellart, "Enhanced transformer model for data-to-text generation," in *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019, Hong Kong, November 4, 2019*, pp. 148–156.

[127] T. Wang, X. Wan, and S. Yao, "Better amr-to-text generation with graph structure reconstruction," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pp. 3919–3925.

[128] L. Song, A. Wang, J. Su, Y. Zhang, K. Xu, Y. Ge, and D. Yu, "Structural information preserving for graph-to-text generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 7987–7998.

[129] T. Liu, F. Luo, P. Yang, W. Wu, B. Chang, and Z. Sui, "Towards comprehensive description generation from factual attribute-value tables," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28-August 2, 2019, Volume 1: Long Papers*, pp. 5985–5996.

[130] F. Nie, H. Chen, J. Wang, R. Pan, and C. Lin, "Beyond word for word: Fact guided training for neural data-to-document generation," in *Natural Language Processing and Chinese Computing - 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9-14, 2019, Proceedings, Part I*, vol. 11838, pp. 525–538.

[131] R. Tian, S. Narayan, T. Sellam, and A. P. Parikh, "Sticking to the facts: Confident decoding for faithful data-to-text generation," *CoRR*, vol. abs/1910.08684, 2019.

[132] C. Rebuffel, L. Soulier, G. Scoutheeten, and P. Gallinari, "Parenting via model-agnostic reinforcement learning to correct pathological behaviors in data-to-text generation," in *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pp. 120–130.

[133] S. Ghosh, Z. Qi, S. Chaturvedi, and S. Srivastava, "How helpful is inverse reinforcement learning for table-to-text generation?" in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pp. 71–79.

[134] R. Puduppully, Y. Fu, and M. Lapata, "Data-to-text generation with variational sequential planning," *Trans. Assoc. Comput. Linguistics*, vol. 10, pp. 697–715, 2022.

[135] Y. Zhu, J. Wan, Z. Zhou, L. Chen, L. Qiu, W. Zhang, X. Jiang, and Y. Yu, "Triple-to-text: Converting RDF triples into high-quality natural languages via optimizing an inverse KL divergence," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pp. 455–464.

[136] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

[137] W. Chen, J. Tian, Y. Li, H. He, and Y. Jin, "De-confounded variational encoder-decoder for logical table-to-text generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pp. 5532–5542.

[138] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pp. 2672–2680.

[139] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pp. 2852–2858.

[140] K. Lin, D. Li, X. He, M. Sun, and Z. Zhang, "Adversarial ranking for language generation," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 3155–3165.

[141] W. Fedus, I. J. Goodfellow, and A. M. Dai, "Maskgan: Better text generation via filling in the _______," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

[142] T. Nishino, R. Ozaki, Y. Momoki, T. Taniguchi, R. Kano, N. Nakano, Y. Tagawa, M. Taniguchi, T. Ohkuma, and K. Nakamura, "Reinforcement learning with imbalanced dataset for data-to-text medical report generation," in *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, pp. 2223–2236.

[143] M. Mager, R. F. Astudillo, T. Naseem, M. A. Sultan, Y. Lee, R. Florian, and S. Roukos, "Gpt-too: A language-model-first approach for amr-to-text generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 1846–1852.

[144] Z. Chen, H. Eavani, W. Chen, Y. Liu, and W. Y. Wang, "Few-shot NLG with pre-trained language model," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 183–190.

[145] H. Gong, Y. Sun, X. Feng, B. Qin, W. Bi, X. Liu, and T. Liu, "Tablegpt: Few-shot table-to-text generation with table structure reconstruction and content matching," in *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pp. 1978–1988.

[146] H. Harkous, I. Groves, and A. Saffari, "Have your text and use it too! end-to-end neural data-to-text generation with semantic fidelity," in *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pp. 2410–2424.

[147] W. Zhao, Y. Liu, Y. Wan, and P. Yu, "Attend, memorize and generate: Towards faithful table-to-text generation in few shots," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, Nov. 2021, pp. 4106–4117.

[148] M. Kale and S. Roy, "Machine translation pre-training for data-to-text generation - A case study in czech," in *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pp. 91–96.

[149] L. F. R. Ribeiro, M. Schmitt, H. Schütze, and I. Gurevych, "Investigating pretrained language models for graph-to-text generation," *CoRR*, vol. abs/2007.08426, 2020.

[150] Q. Guo, Z. Jin, X. Qiu, W. Zhang, D. Wipf, and Z. Zhang, "Cyclegt: Unsupervised graph-to-text and text-to-graph generation via cycle training," *CoRR*, vol. abs/2006.04702, 2020.

[151] M. Kale and A. Rastogi, "Text-to-text pre-training for data-to-text tasks," in *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pp. 97–102.

[152] A. M. Hoyle, A. Marasovic, and N. A. Smith, "Promoting graph awareness in linearized graph-to-text generation," in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, pp. 944–956.

[153] Q. Wang, S. Yavuz, X. V. Lin, H. Ji, and N. F. Rajani, "Stage-wise fine-tuning for graph-to-text generation," in *Proceedings of the ACL-IJCNLP 2021 Student Research Workshop, ACL 2021, Online, JUli 5-10, 2021*, pp. 16–22.

[154] X. Xing and X. Wan, "Structure-aware pre-training for table-to-text generation," in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, pp. 2273–2278.

[155] J. Li, T. Tang, W. X. Zhao, Z. Wei, N. J. Yuan, and J. Wen, "Few-shot knowledge graph-to-text generation with pretrained language models," in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, pp. 1558–1568.

[156] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pp. 4582–4597.

[157] J. Clive, K. Cao, and M. Rei, "Control prefixes for text generation," *CoRR*, vol. abs/2110.08329, 2021.

[158] Y. Luo, M. Lu, G. Liu, and S. Wang, "Few-shot table-to-text generation with prefix-controlled generator," in *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pp. 6493–6504.

[159] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *CoRR*, vol. abs/2003.08271, 2020.

[160] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pp. 2227–2237.

[161] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 3079–3087.

[162] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pp. 2873–2879.

[163] B. McCann, J. Bradbury, C. Xiong, and R. Socher, "Learned in translation: Contextualized word vectors," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6294–6305.

[164] J. Devlin, M. Chang, K. Lee, and 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186.

[165] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 5754–5764.

[166] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019.

[167] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, and H. Hon, "Unified language model pre-training for natural language understanding and generation," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 13 042–13 054.

[168] H. Bao, L. Dong, F. Wei, W. Wang, N. Yang, X. Liu, Y. Wang, J. Gao, S. Piao, M. Zhou, and H. Hon, "Unilmv2: Pseudo-masked language models for unified language model pre-training," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, vol. 119, pp. 642–652.

[169] A. Radford and K. Narasimhan, "Improving language understanding by generative pre-training," 2018.

[170] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[171] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[172] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 7871–7880.

[173] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, pp. 140:1–140:67, 2020.

[174] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, "Multilingual denoising pre-training for neural machine translation," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 726–742, 2020.

[175] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pp. 483–498.

[176] K. Song, X. Tan, T. Qin, J. Lu, and T. Liu, "MASS: masked sequence to sequence pre-training for language generation," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, vol. 97, pp. 5926–5936.

[177] Z. Chi, L. Dong, F. Wei, W. Wang, X. Mao, and H. Huang, "Cross-lingual natural language generation via pre-training," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 7570–7577.

[178] M. Schmitt, S. Sharifzadeh, V. Tresp, and H. Schütze, "An unsupervised joint system for text generation from knowledge graphs and semantic parsing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 7117–7130.

[179] X. Yin and X. Wan, "How do seq2seq models perform on end-to-end data-to-text generation?" in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 7701–7710.

[180] E. Chang, X. Shen, D. Zhu, V. Demberg, and H. Su, "Neural data-to-text generation with lm-based text augmentation," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pp. 758–768.

[181] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *CoRR*, vol. abs/2107.13586, 2021.

[182] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," 2021.

[183] J. Qian, L. Dong, Y. Shen, F. Wei, and W. Chen, "Controllable natural language generation with contrastive prefixes," in *Findings of the Association for Computational Linguistics: ACL 2022*, Dublin, Ireland, May 2022, pp. 2912–2924.

[184] Y. Zhao, H. Zhang, S. Si, L. Nan, X. Tang, and A. Cohan, "Large language models are effective table-to-text generators, evaluators, and feedback providers," *CoRR*, vol. abs/2305.14987, 2023.

[185] T. C. Ferreira, C. van der Lee, E. van Miltenburg, and E. Krahmer, "Neural data-to-text generation: A comparison between pipeline and end-to-end architectures," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, November 3-7, 2019*, pp. 552–562.

[186] C. Kedzie and K. R. McKeown, "Controllable meaning representation to text generation: Linearization and data augmentation strategies," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 5160–5185.

[187] L. Li and X. Wan, "Point precisely: Towards ensuring the precision of data in generated texts using delayed copy mechanism," in *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pp. 1044–1055.

[188] S. Ma, P. Yang, T. Liu, P. Li, J. Zhou, and X. Sun, "Key fact as pivot: A two-stage model for low resource table-to-text generation," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 2047–2057.

[189] L. Miculicich, M. Marone, and H. Hassan, "Selecting, planning, and rewriting: A modular approach for data-to-document generation and translation," in *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019, Hong Kong, November 4, 2019*, pp. 289–296.

[190] X. Shen, E. Chang, H. Su, C. Niu, and D. Klakow, "Neural data-to-text generation via jointly learning the segmentation and

correspondence," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 7155–7165.

[191] Z. Li, Z. Lin, N. Ding, H. Zheng, and Y. Shen, "Triple-to-text generation with an anchor-to-prototype framework," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pp. 3780–3786.

[192] H. Gong, W. Bi, X. Feng, B. Qin, X. Liu, and T. Liu, "Enhancing content planning for table-to-text generation with data understanding and verification," in *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, pp. 2905–2914.

[193] Z. Kasner and O. Dusek, "Data-to-text generation with iterative text editing," in *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pp. 60–67.

[194] K. Chen, F. Li, B. Hu, W. Peng, Q. Chen, H. Yu, and Y. Xiang, "Neural data-to-text generation with dynamic content planning," *Knowl. Based Syst.*, vol. 215, p. 106610, 2021.

[195] R. Puduppully and M. Lapata, "Data-to-text generation with macro planning," *Trans. Assoc. Comput. Linguistics*, vol. 9, pp. 510–527, 2021.

[196] Y. Su, D. Vandyke, S. Wang, Y. Fang, and N. Collier, "Plan-then-generate: Controlled data-to-text generation via planning," in *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pp. 895–909.

[197] P. Wang, J. Lin, A. Yang, C. Zhou, Y. Zhang, J. Zhou, and H. Yang, "Sketch and refine: Towards faithful and informative table-to-text generation," in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, vol. ACL/IJCNLP 2021, pp. 4831–4843.

[198] C. van der Lee, E. Krahmer, and S. Wubben, "Automated learning of templates for data-to-text generation: comparing rule-based, statistical and neural methods," in *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pp. 35–45.

[199] A. Upadhyay, S. Massie, R. K. Singh, G. Gupta, and M. Ojha, "A case-based approach to data-to-text generation," in *Case-Based Reasoning Research and Development - 29th International Conference, ICCBR 2021, Salamanca, Spain, September 13-16, 2021, Proceedings*, ser. Lecture Notes in Computer Science, vol. 12877, pp. 232–247.

[200] S. Jolly, Z. X. Zhang, A. Dengel, and L. Mou, "Search and learn: Improving semantic coverage for data-to-text generation," in *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pp. 10 858–10 866.

[201] Y. Uehara, T. Ishigaki, K. Aoki, H. Noji, K. Goshima, I. Kobayashi, H. Takamura, and Y. Miyao, "Learning with contrastive examples for data-to-text generation," in *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*. International Committee on Computational Linguistics, 2020, pp. 2352–2362.

[202] L. Perez-Beltrachini and C. Gardent, "Analysing data-to-text generation benchmarks," in *Proceedings of the 10th International Conference on Natural Language Generation, INLG 2017, Santiago de Compostela, Spain, September 4-7, 2017*, 2017, pp. 238–242.

[203] S. Shen, D. Fried, J. Andreas, and D. Klein, "Pragmatically informative text generation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1*, pp. 4060–4067.

[204] W. Li, W. Wu, M. Chen, J. Liu, X. Xiao, and H. Wu, "Faithfulness in natural language generation: A systematic survey of analysis, evaluation and optimization methods," *CoRR*, vol. abs/2203.05227, 2022.

[205] Y. Ma, K. L. Nguyen, F. Z. Xing, and E. Cambria, "A survey on empathetic dialogue systems," *Inf. Fusion*, vol. 64, pp. 50–70, 2020.

[206] S. Lin, W. Wang, Z. Yang, X. Liang, F. F. Xu, E. Xing, and Z. Hu, "Data-to-text generation with style imitation," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online, Nov. 2020, pp. 1589–1598.

[207] Y. Lin, T. Ruan, M. Liang, T. Cai, W. Du, and Y. Wang, "Dotat: A domain-oriented text annotation tool," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL 2022 - System Demonstrations, Dublin, Ireland, May 22-27, 2022*, pp. 1–8.

[208] L. Dou, G. Qin, J. Wang, J. Yao, and C. Lin, "Data2text studio: Automated text generation from structured data," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pp. 13–18.

[209] Z. Kasner, E. Garanina, O. Plátek, and O. Dusek, "Tabgenie: A toolkit for table-to-text generation," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2023, Toronto, Canada, July 10-12, 2023*, 2023, pp. 444–455.

Yupian Lin is a Ph.D. student with the School of Information Science and Engineering, East China University of Science and Technology, Shanghai, China. His research interests include data-to-text generation, language modeling, and question answering.

Tong Ruan received the Ph.D. degree in computer technology from Institute of Software, Chinese Academy of Science. She is a professor of artificial intelligence at East China University of Science and Technology, Shanghai, China. Her research interests include knowledge graph, natural language processing and clinical informatics.

Jingping Liu received the Ph.D. degree from the School of Computer Science, Fudan University, China, in 2021. He now is a lecturer at the School of Information Science and Engineering, East China University of Science and Technology, China. His research interests include commonsense reasoning, knowledge bases, and data mining.

Haofen Wang received the Ph.D. degree in computer application from Shanghai Jiaotong University. He is a professor of artificial intelligence at Tongji University, Shanghai, China. His research interests include knowledge graph, data mining, and natural language processing.