Review:

Advances and challenges in artificial intelligence text generation[*]

Bing LI[†1,2], Peng YANG[†‡1,2], Yuankang SUN[†1,2], Zhongjian HU[†1,2], Meng YI[1,2]

¹School of Computer Science and Engineering, Southeast University, Nanjing 210000, China

²Key Laboratory of Computer Network and Information Integration, Ministry of Education,

Southeast University, Nanjing 210000, China

[†]E-mail: libing@seu.edu.cn; pengyang@seu.edu.cn; syk@seu.edu.cn; huzj@seu.edu.cn

Received June 9, 2023; Revision accepted Sept. 18, 2023; Crosschecked Jan. 2, 2024

Abstract: Text generation is an essential research area in artificial intelligence (AI) technology and natural language processing and provides key technical support for the rapid development of AI-generated content (AIGC). It is based on technologies such as natural language processing, machine learning, and deep learning, which enable learning language rules through training models to automatically generate text that meets grammatical and semantic requirements. In this paper, we sort and systematically summarize the main research progress in text generation and review recent text generation papers, focusing on presenting a detailed understanding of the technical models. In addition, several typical text generation application systems are presented. Finally, we address some challenges and future directions in AI text generation. We conclude that improving the quality, quantity, interactivity, and adaptability of generated text can help fundamentally advance AI text generation development.

Key words: AI text generation; Natural language processing; Machine learning; Deep learning

https://doi.org/10.1631/FITEE.2300410   **CLC number:** TP181

1 Introduction

With the rapid development of deep learning technology, artificial intelligence (AI) text generation performance has significantly improved. As an important supporting technology for natural language, it has been extended from the initial field of text generation to multiple application scenarios such as machine translation, question-and-answer (Q&A) systems, speech recognition, natural language understanding, and multimodal interaction. It has ex-

tremely important research and application value in both academia and industry (Garbacea and Mei, 2020).

Intelligent text generation refers to automatically generating various high-quality text types with good readability based on different input data types. At the early stage of AI development, text generation was mostly based on artificially formulated rules or templates due to various limiting factors, which is not essentially intelligent creativity (Wang WM et al., 2017). In recent years, deep learning based algorithms have enabled intelligent text generation models to overcome the limitations of regularization and templating completely, so text can be generated quickly and flexibly based on different data modalities (See et al., 2017). In other words, the rise of intelligent text generation stems from breakthroughs in deep learning technology and the exponential growth

---

ORCID: Bing LI, https://orcid.org/0000-0002-1251-4346; Peng YANG, https://orcid.org/0000-0002-1184-8117

of data resources, both of which satisfy the learning requirements of the models. In addition, with the advent of the digital era, the demand for intelligent text generation has increased in major industries (Omar et al., 2023).

As shown in Fig. 1, traditional content production and distribution cannot meet consumer needs in various industries due to human resource limitations. While decision-based content production improves mainly the content distribution efficiency, it still does not eliminate the limitation of manually produced content. Generative content production and distribution enables the utilization of AI for creation and production, changing the traditional manner in which content is produced by humans. Therefore, supported by massive demand, platform giants such as Google, Microsoft, Alibaba, and ByteDance have invested significant resources to conduct research and development in intelligent text generation to meet the considerable demand in the digital market, thus enhancing productivity.

The increased performance of hardware devices and the proliferation of large-scale data are the fundamental reasons for the rapid development of AI technologies. Major companies have started investing massive amounts of energy and financial resources in researching AI generation models. In intelligent text generation, the most representative

products include ChatGPT (Lund and Wang, 2023) created by OpenAI, LaMDA designed by Google for Q&A (Thoppilan et al., 2022), Turing-NLG, a mature product developed by Microsoft for summary generation and text translation (Smith et al., 2022), and ERNIE, a product invented by Baidu (Zhang ZY et al., 2019).

Among them, the generative pre-trained Transformer (GPT) (Floridi and Chiriatti, 2020) family of models introduced by OpenAI is the most successful large text generation model thus far. The model employs a conversational format to interact with people and can answer follow-up questions, acknowledge errors, question incorrect premises, and reject inappropriate requests (King and ChatGPT, 2023). ChatGPT exhibits powerful capabilities not only in everyday conversation, professional question answering, information retrieval, content continuation, literary composition, and music composition, but also in generating and debugging code and generating comments (Surameery and Shakor, 2023). The emergence of ChatGPT represents a significant advancement in AI technology for customer service and support. It addresses the increasing demand and expectations for intelligent and virtual assistants. It also demonstrates the growing interests and desires of human society in the pursuit of intelligence and automation. The emergence of ChatGPT is likely
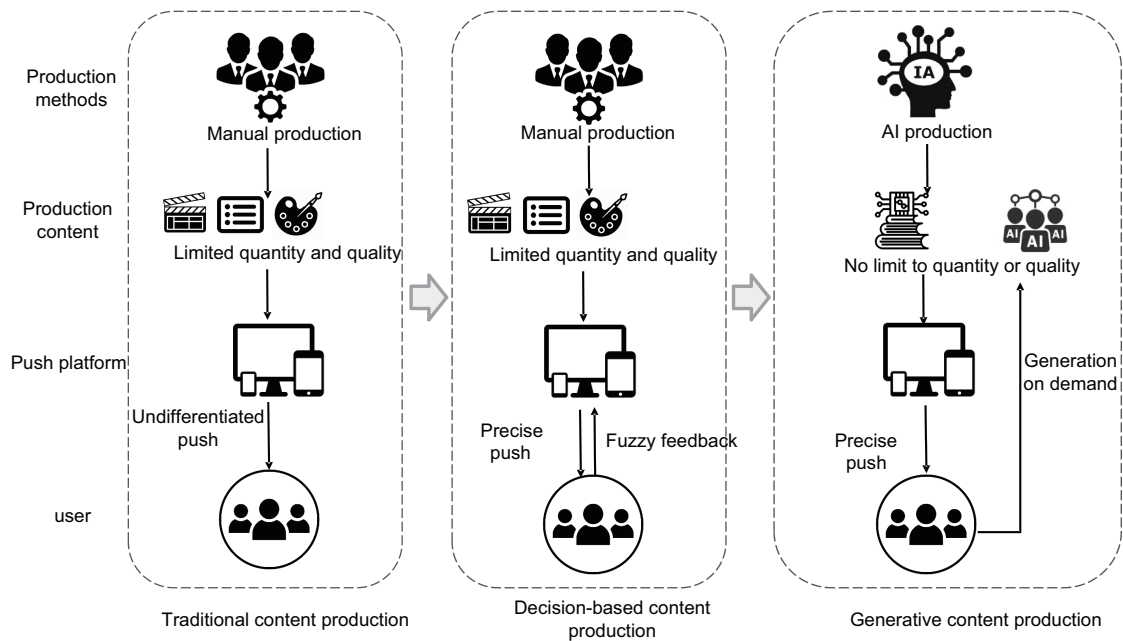

Fig. 1 Different content production methods

to signify the emergence of a new customer support model that creates more business value for companies by improving customer satisfaction and experience through technologies such as deep learning algorithms.

Concurrently, the emergence of ChatGPT has resulted in massive content generation, making the governance of content indispensable. The most popular research at present is to detect the authenticity of content through fact-checking (Zhao GZ and Yang, 2022; Procter et al., 2023). Uniform content label (UCL) is one of the competitive solutions for AI-generated content (AIGC) governance, which was proposed by Professor Youping LI, the academician of the Chinese Academy of Engineering. With standardized content identification and dual-signature mechanism, as well as the entanglement and consistency in multiple dimensions such as semantics, time, and space, UCL can ensure the security, trustworthiness, and traceability of AIGC.

To increase future researchers' systematic understanding of the directions and cutting-edge technological developments in intelligent text generation, in this paper, we summarize some specific manifestations of intelligent text generation. In addition, research advances in the field are outlined, and the challenges facing intelligent text generation are highlighted. Finally, we provide several valuable directions for future research development. We conclude that exploring lightweight models to improve intelligent text generation quality will facilitate the rapid development of the intelligent text industry chain. The main contributions of this paper are as follows:

1. We summarize the main forms of AI text generation specifically, highlighting the exploration of three forms of text generation and their representative task forms.

2. We discuss the technical research progress in AI text generation in the past eight years and introduce representative papers exploring the performance advantages of several large-scale pre-trained models in intelligent text generation.

3. We provide comprehensive and integrated research analysis and summary of the current state of intelligent text applications and introduce some products with mature applications from a practical perspective.

4. Finally, we analyze the challenges and propose several valuable research trends for the future.

2 Main forms of text generation

Text generation is the procedure of producing data through certain AI algorithms. Fig. 2 shows the text generation development stages. We can see that there are four main text generation stages, namely, professionally generated content (PGC), user-generated content (UGC), AI-assisted generated content (AIAGC), and AIGC, the first three of which are limited by human manufacturing ability, imagination, knowledge level, and other factors, resulting in restrictions on the quality and quantity of generated text. In contrast, AIGC is completely free of such restrictions, ensuring the quality, efficiency, and freedom of generated text (Senftleben et al., 2020; Jungwirth and Haluza, 2023).
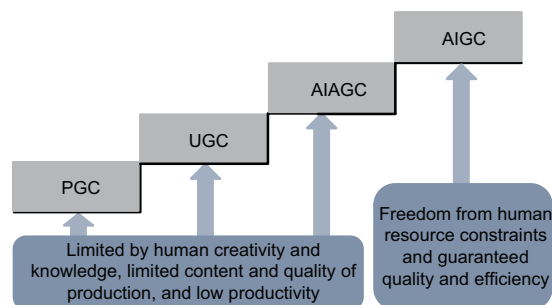

Fig. 2 Four stages of content generation

Text generation in AIGC can be divided into the following three main categories according to the content and characteristics:

1. Utility text generation. Computer technology and natural language processing techniques are used to automatically generate utility text content based on specific application scenarios and user requirements, such as generating news (Leppänen et al., 2017), reports (Chen ZH et al., 2021), product manuals, operation manuals (Zogopoulos et al., 2022), and business contracts. Text generation is generally provided by structured data or predefined rules and templates. The generated text must follow certain formats and standards. For example, the system automatically fills in various terms, amounts, times, and other information to generate a commercial contract that conforms to legal standards and industry norms. With an emphasis on practicality and operability, these kinds of text generation technologies provide a more efficient and convenient solution for information processing and are essentially an automated text-writing process (Guo et al., 2023). Utility

text generation is characterized by efficiency, precision, and customizability. The common utility text generation tasks include the following:

(1) Text summary. A text summary is the automatic analysis of a text by technology to compress a longer text or article into a shorter, representative summary text. The sentences of the abstract are either from the source text or rewritten from the original content. The ultimate purpose is to provide the key viewpoints and information of the text, which is extremely valuable for long texts such as news, research papers, and books (Li B et al., 2023). News headline generation, keyword generation, dissertation abstract generation, and book introduction generation all belong to the text summary category (Zhu QH and Luo, 2023).

(2) Content generation. Content is one of the most significant Internet resources, and automated content generation has attracted the attention of a growing number of researchers. Using text generation techniques to produce various content, such as articles, news reports, and recommended messages, can reduce labor costs and improve production efficiency (Oh et al., 2020).

(3) Intelligent report generation. Reports require synthesizing multiple data sources, conducting intensive analysis, and generating structured reports. Text generation, such as business report generation and financial statement generation, can automatically generate simple descriptive language to enhance the clarity and understandability of figures and reports.

(4) Automated text generation by speech. This technology uses mainly a combination of text-to-speech (TTS) and automatic speech recognition (ASR) technologies to convert text to speech or speech to text (Trivedi et al., 2018). It can be applied to automatically recognize voice content and convert it into text format and can also be employed to automatically convert text into voice. It is mostly available in voice assistants, natural language processing, and intelligent homes.

2. Creative text generation. By using AI technology, we generate creative texts by learning and imitating massive text data to realize creative work with abundant imagination, including autonomous poetry composition, story creation, and songwriting (Saeed et al., 2019). Creative text generation not only has the potential to produce valuable works of literature and art but also contributes to broadening the creative discoverability of artificial literary and artistic works, thus promoting innovative thought processes. Creative text generation provides applications in a variety of fields, including but not limited to the following:

(1) Literary and artistic creation. Creative text generation enables the generation of creative and artistic literary works, such as poetry, novels, and essays (Tian and Peng, 2022). Using creative text generation technology stimulates the imagination and creativity of human literary creation and drives the overall development of literary art activities.

(2) Advertising e-commerce text creation. Creative text generation produces creative and attractive advertising or marketing words based on human requirements and objectives, and provides more pertinent content for brand advertising, short video writing, and reliable copywriting (Zhang XY et al., 2022a, 2022b).

(3) Social media text creation. In modern social media communication, creative text generation helps people express ideas and thoughts rapidly and precisely by making full use of the advantages and power of social media platforms (Cao and Wang, 2018).

3. Emotive text generation. In this type of text generation, computers are used to simulate the expression of human emotions, and natural language processing and machine learning techniques are employed to construct models that generate texts that match distinct emotional styles. The goal of emotive text generation is to produce texts that express emotions naturally and accurately. The generated texts include those that describe emotions, reactions, emotional states, or opinions. Understanding the semantics and structure of emotion is crucial to generating emotive texts. This requires identifying and understanding sentiment words, sentence structure, context, and tone, generating meaningful text (Zhang YQ and Huang, 2019; Birjali et al., 2021). To achieve sentiment text generation, a variety of computer vision and natural language processing techniques are needed, such as deep learning, text classification, sequence-to-sequence models (Fei et al., 2022), and natural language generation. The main application scenarios of sentiment text generation are sentiment analysis (Mukherjee, 2021), chatbots (Adamopoulou and Moussiades, 2020), email and social text writing, and product evaluation.

From the above analysis, we can conclude that utility text generation, creative text generation, and emotive text generation have different characteristics. We summarize these characteristics in Table 1, specifically:

1. Utility text generation is typically employed to produce texts with practical value, such as e-commerce details, letters, and reports. Creative text generation places more emphasis on generating creative and personalized texts, such as novels, news reports, and poems. Emotive text generation focuses on emotional diversity, such as rich expressions and emotional categories.

2. Utility text generation relies on many existing standardized and normalized datasets and knowledge systems, creative text generation focuses on exploring aspects including language and emotion, and emotive text generation explores user emotions and feedback through algorithmic models and sentiment-like analysis techniques.

3. Utility text generation primarily updates delivery and allows high-speed fulfillment of various tasks, creative text generation concentrates on assisting creative inspiration and generating data for human work, and emotive text generation emphasizes digging deeper into the emotional experience of users to provide a higher added value for company decisions.

3 Research progress in text generation

Text generation is an essential research problem in AI, and significant progress has been achieved in the text generation domain in recent years with the continuous advancement of technology. According to the technical solutions implemented, they can be divided into rule-based, statistic-based, deep learning based, and hybrid text generation methods.

3.1 Rule-based text generation methods

Earlier approaches to text generation are primarily rule-based methods. These methods lever-

age predefined rules and templates to generate text. Their essence is to automatically generate text by translating predefined rules and logic into information that computers can understand and process. These rules and logic range from complex grammar, sentence structure, and vocabulary usage rules to simple templates and placeholders. The approach allows control over the content and form of the generated text and is, therefore, commonly employed to generate structured text, such as forms, reports, and contracts.

The most representative approach is based on predefined text templates and rules for generating text containing placeholders and conditional statements. This category of methods fills or excludes blocks or segments of text based on the generated information and is, therefore, more suitable for generating structured text, including emails, resumes, and business letters.

In addition, various researchers use specific grammar rules to generate text, such as context-free grammar (CFG) (Yellin and Weiss, 2021) and regular expressions (Zheng et al., 2021). These rules restrict the structure and format of the generated text and, therefore, tend to generate more accurate and uniform text, including speech to text (Trivedi et al., 2018; Wang CH et al., 2020) and code generation (Svyatkovskiy et al., 2020; Li YJ et al., 2022). Tran and Bui (2021) proposed recognizing speech-to-text sequences using rich features and manually designed rules for their normalization. Imam et al. (2014) introduced an expert code generator based on rule and framework knowledge representation techniques. A rule-based fusion of knowledge and reasoning strategies for text generation in certain specialized domains, such as medical diagnosis and legal documents, is needed. Kraus et al. (2016) generated critical care discharge letters using Arden syntax.

The rule-based approach ensures the generated text accuracy and consistency, so it is convenient to preserve and manage, and the generation speed is high with certain controllability and predictability

Table 1 Comparison of the characteristics of three different types of text generation

Category	Efficient	Customizability	Innovative	Artistry	Adaptive	Diversity
Utility text generation	Yes	Yes	No	No	Yes	Yes
Creative text generation	Yes	No	Yes	Yes	Yes	Yes
Emotive text generation	Yes	Yes	Yes	No	Yes	Yes

(Croft et al., 2021). Because deep learning and other techniques cannot train or adjust in real time, the rule-based approach suffers from restricted generated result quality.

3.2 Statistic-based text generation methods

Statistical-based text generation methods analyze the word frequency, sentence length, paragraph structure, and document frequency information in text, employ machine learning methods to calculate the value of each sentence, and then select a certain number of sentence compositions to generate the text. The representative algorithms include frequency statistics (Zhao W et al., 2019) and sentence position (Xu et al., 2021).

3.2.1 Frequency-based text generation method

The frequency-based text generation method involves the researcher empirically determining which words can be included in the final generation based on how often the words appear in the text, excluding deactivated words (Vodolazova and Lloret, 2019). The specific steps are shown in Fig. 3. First, the original text is transformed into a series of words or phrases, and operations such as word form reduction and the removal of stop words are performed to improve the text representation. Then, the number of times each word or phrase appears in the text is counted. A weight is assigned to each word or phrase based on its occurrence frequency in the text. The term frequency–inverse document frequency (TF–IDF) (Christian et al., 2016) method is typically employed to calculate weights. The higher the word frequency, the higher the corresponding weight, and the sentence weight is the sum of the weights of the

words that compose the sentence. Finally, the most important keywords are selected from the words or phrases with high weights as part of the generated text.

The word frequency statistic method is understandable, computationally efficient, and applied to text summarization and content generation for utility text generation. Luo et al. (2016) proposed a text keyword extraction method based on word frequency statistics. The word frequency in the text was determined by the formula of the number of words with the same frequency, and the frequency statistics were applied to keyword extraction. He et al. (2022) used statistics on the frequency of candidate watermarked words to design a conditional watermarking framework to protect the intellectual property of the text generation application programming interface (API). The word frequency based text generation method is convenient to implement and is computationally efficient. However, it fails to consider the semantic word information and tends to ignore semantically important but infrequently occurring words; thus, it cannot consider the contextual background, resulting in the loss of key information.

3.2.2 Sentence position based text generation method

The sentence position based text generation method is a classical technique whose fundamental ideology is to select numerous key sentences according to their positions in the text to compose the final generated text. The following methods are commonly applied: position-based scoring, where sentences at the beginning and end of the text are considered the most important and thus higher weights
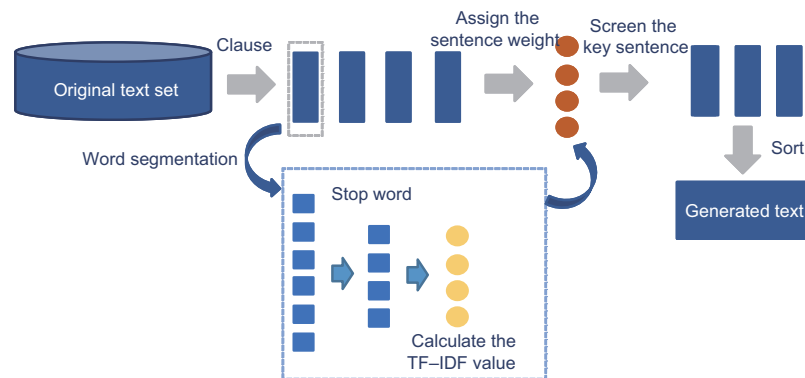

Fig. 3 Breakdown of the methodological steps of word frequency statistics

are given to these sentences, and sentence similarity based scoring, which considers that sentences with higher similarity to other sentences are more frequently selected to be included in the generated text and thus higher weights are assigned to the sentences. This kind of sentence position based method is effective and convenient to implement, and it works well for some texts containing obvious time and event relationships, such as news and technology.

Lead3 is a classical approach for obtaining generated text based on sentence position; it is mostly applied in news summaries, where the opening section of the default article usually contains the most important information (Koupaee and Wang, 2018). Barrera and Verma (2012) combined syntactic, grammatical, and statistical strategies to design specific selection patterns to construct news summaries. Sun et al. (2022) argued that sentence similarity should be measured by comparing the probability of generating two sentences in the same context. Özateş et al. (2016) proposed a sentence similarity calculation method based on sentence dependency grammar representation and applied it in extractive multidocument summarization.

However, text generation methods based on sentence position fail to smoothly resolve the deep semantic relationships between sentences, resulting in a lack of coherence in the generated text. In addition, these methods are easily disrupted by noisy materials, and the generated texts are unsatisfactory in terms of stability and accuracy of effects.

3.3 Deep learning based text generation methods

The rapid development of deep learning has attracted the attention of many researchers; we identify the research papers on deep learning in the last eight years in Fig. 4, which demonstrates that deep learning techniques are favored by major researchers. Emerging deep learning techniques have also revolutionized text summarization technologies. Their continuous iterations in learning paradigms and network structures have greatly improved the learning capabilities of AI algorithms, thus driving the rapid development of intelligent text technologies. Deep learning based approaches allow text processing with more sophisticated models, such as convolutional neural networks (CNNs) (Albawi et al., 2017), long short-term memory (LSTM) networks
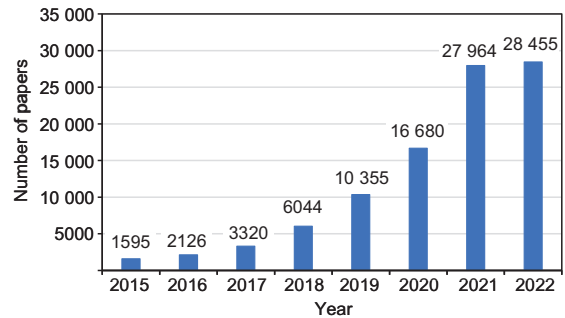
Fig. 4 Statistics on the number of articles published on deep learning in the past eight years

(Palangi et al., 2016), and Transformers (Vaswani et al., 2017), which provide a better understanding of the semantic and contextual information of text and generate more accurate and coherent generated text.

Deep learning based text generation was first proposed by Kingma and Welling (2014) with a deep variational self-encoder, which enables reconstruction and parameterized network learning based on the input, greatly improving the technical progress of intelligent text generation. With the proposal of generative adversarial networks (GANs) (Goodfellow et al., 2020), generated text quality has been significantly improved by generators and discriminators continuously playing and learning between the generated and real data. With the structural upgrade of deep neural networks, increasingly large-scale neural networks have been proposed to be applied to intelligent text. The most representative network is a framework based on end-to-end processing, i.e., data input–deep learning model–data output. Among deep learning models, the most representative methods are encoder–decoder-based text generation methods and pre-trained language model based text generation methods.

3.3.1 Encoder–decoder-based text generation method

The method is divided into two components: the encoder is applied to the embedded representation and encoding of the input data to calculate the representation of semantic vector text features, and the decoder provides the encoder output as the input to decode the word sequences for output. For the encoder selection, the input sequence is encoded and represented by CNNs, recurrent neural networks (RNNs) (Yu Y et al., 2019), LSTM networks, and

Transformers. The decoder employs RNNs, LSTM networks, and Transformers to decode the output representation of word sequences for text generation tasks. In principle, the encoder and decoder can be a combination of any of the above structures; however, in practice, the choices of encoder and decoder structures are basically the same.

This approach, where both the input and output are sequences, is collectively referred to as the Seq2Seq model; the specific architecture is shown in Fig. 5. The input sequence is encoded by the encoder after embedding, and then the output is decoded by the decoder according to the hidden state of the encoded output. The <BOS> and <EOS> in Fig. 5 are the start and end symbols, respectively. In the decoding process, the attention mechanism is added to Seq2Seq to assist the model in giving precise attention to the information-rich inputs, and the attention weight coefficients are calculated based on the correlation between the current decoding state and each encoder state, with the contextual vector representation of the input information at the current decoding moment obtained based on the weight coefficients (Yu X et al., 2019). As shown in Fig. 5, assuming that in decoding the symbol Y, the model pays more attention to the coded representation of the input sequence B, which means that less attention is given to the representation of the other input sequences, sequence B plays a decisive role in generating the decoded Y. Two classically different encoder–decoder models are highlighted below:

1. Seq2Seq model based on neural networks. The encoder and decoder of this model choose the neural network. Neural networks include CNNs, RNNs, gated recurrent units (GRUs) (Dey and Salem, 2017), and LSTM networks, in which RNNs belong to the basic network. LSTM is a special form of network based on an RNN that perfectly allevi-

ates gradient disappearance and gradient explosion problems. GRU is a variant of LSTM, and the structure is comparatively simpler than that of LSTM. In terms of encoder and decoder selection, LSTM allows unidirectional encoding and decoding, but such unidirectional encoding fails to facilitate the model to leverage the backward information. Therefore, in practical research and applications, most researchers prefer to deepen the RNN layers or add a backward RNN, such as Bi-LSTM, to exploit the forward and backward coding information. In addition, some researchers have added attention mechanisms to the model to improve the model-generated sequence accuracy. Table 2 presents a summary of some representative papers in recent years.

2. Seq2Seq model based on the Transformer. The Transformer model encoder consists of multihead attention, add & norm, and feed forward, as shown in Fig. 6. The decoder includes an additional masked multihead attention module on top of the encoder. The encoder obtains a context-aware semantic vector representation for each input text, and the decoder layer assigns weights to the encoder output representation by applying multiheaded attention, focusing on the information-rich semantic representation while using steganographic multiheaded attention to the generated sequence information. Transformers are a breakthrough technology in natural language processing (NLP) and have superior performance on various text generation tasks. Table 3 summarizes some representative Transformer-based Seq2Seq text generation papers in recent years.

The two Seq2Seq models mentioned above perform well in text generation tasks. Many researchers have proposed generation models with superior performance, such as replacing encoders and decoders, adding attention mechanisms according to specific
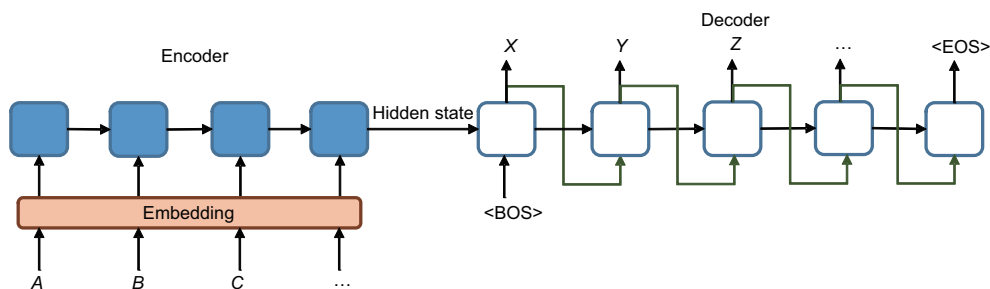

Fig. 5 Schematic of the Seq2Seq model

Table 2 Introduction of representative papers on the Seq2Seq model based on neural networks

Model	Description
SAN-T2T (Ding and Xu, 2023)	The generative model consists of a field content selective encoder and a descriptive decoder with the addition of a selective attention network to determine the importance of the fields during the encoding phase
HSIG (Li B et al., 2023)	HSIG uses Bi-LSTM to introduce hierarchical information reasoning between questions and corresponding answers to form inferable and interpretable abstracts
GGNANs (Liang et al., 2021)	A gated-graph neural attention network is proposed. It is fused with the Seq2Seq model to encode global information and to obtain extractable summaries with a superior performance
PAAG (Gao et al., 2019)	The model distinguishes facts using a CNN and captures the most relevant words in a given question employing an attention mechanism to generate product-related answers and questions
Field-gating Seq2Seq (Liu TY et al., 2018)	A structure-aware Seq2Seq structure is proposed consisting of a field-gating encoder and a dual-attention description generator, thus encoding the table content and structure in a deep semantic context
Seq2Seq+CGU (Lin JY et al., 2018)	A global encoding framework is proposed to control the information flow from the encoder to the decoder using the global information of the context
PGN (See et al., 2017)	A hybrid pointer-generator network based on Seq2Seq is proposed to prevent duplication by generating new words while keeping track of what was generated by the generator
CopyRNN (Meng et al., 2017)	The model captures the deep semantics of the content using the deep learning approach of the Seq2Seq framework to predict the keyword generation

Bi-LSTM: bidirectional long short-term memory; CNN: convolutional neural network
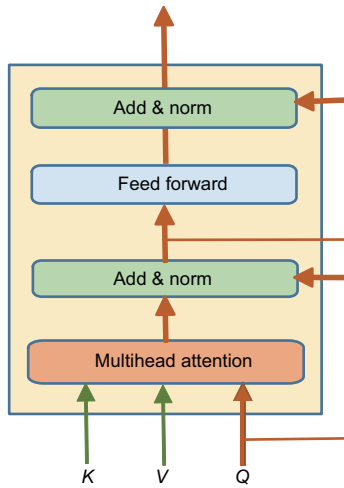
Fig. 6 Encoding structure of the Transformer

tasks, or changing loss strategies, but they are all supervised Seq2Seq models in nature. In addition, variational autoencoders (VAEs) (Tomczak and Welling, 2016) describe the observation of potential space probabilistically and demonstrate a significant application in data generation. VAEs and GANs are regarded as the most researched methods in unsupervised learning and are increasingly used in in-depth text generation models.

3.3.2 Pre-trained language model based text generation method

With the emergence of massive quantities of unlabeled text, language models based on large-scale pre-trained models enable full understanding, mining, and learning of semantic information to learn more parameters. Pre-trained models generally consist of multiple neural network modules stacked on top of each other, and the most commonly applied model is the Transformer network with many parameters and considerable training data, which leads to a complex network structure (King, 2023). Among the pre-trained models that perform well in terms of encoders are the bidirectional encoder representations from Transformers (BERT) (Devlin et al., 2019) and RoBERTa (Liu YH et al., 2019) models. As shown in Fig. 7a, the model inputs are A, C, and E, and the model is required to encode the outputs B and D. This class of models focuses more on the representational learning of coded text. It also validates the superiority of large-scale language models in zero-resource, small-sample, and low-medium-resource scenarios.

Some researchers have applied text generation

Table 3 Introduction of representative papers on the Seq2Seq model based on the Transformer

Model	Description
Transformer+atten (Li JJ et al., 2022)	The Transformer model is introduced into the question generation process, and the attention network model is used to obtain a diverse representation of the question
Graph Transformer (Wang TM et al., 2020)	The paper proposes a new graph-to-sequence model that learns node representation relations by encoding AMR graphs directly and aggregates graph neighbor node information using an attention mechanism
PAIR (Hua and Wang, 2020)	The content-controlled text generation framework PAIR is based on the large-scale model BART for planning and iterative improvement. BERT is employed to automatically construct the content encoding, and the BART model is adopted for information decoding generation
CTRL (Keskar et al., 2019)	The conditional transformation language model CTRL achieves control over the generated text by controlling specific behaviors such as style, content, and task
PL-FT (Zhu CG et al., 2021)	The paper designs a pre-trained abstract news summary model on a large-scale unlabeled corpus using introductory bias to predict introductory statements
GraphWriter (Koncel-Kedziorski et al., 2019)	The paper introduces a new graph transformation encoder that exploits the relational structure of knowledge graphs for encoding representation, combined with the Seq2Seq structure for graph-to-text generation
PPLM (Dathathri et al., 2020)	PPLM for controlled language generation, combines a pre-trained LM with one or more simple attribute classifiers that guide text generation without LM training
HT (Liu Y and Lapata, 2019a)	A hierarchical approach to cross-document relationship extraction based on the Transformer architecture is presented to understand the potential dependencies between text units

AMR: abstract meaning representation; BART: bidirectional and auto-regressive Transformers; BERT: bidirectional encoder representations from Transformers; LM: language model

tasks and achieved decent performance. Liu Y and Lapata (2019b) showed how to use BERT for effective text summary generation. Miller (2019) performed text embedding and K-means clustering using the BERT model to identify sentences near the center of mass for summary selection. Zhang TY et al. (2020) exploited a fine-tuned BERT to supervise an improved traditional Seq2Seq model for improved text generation performance. Chen YC et al. (2020) proposed a conditional mask language modeling (C-MLM) task to implement BERT to fine-tune the target text generation dataset. Zhao W et al. (2022) introduced the parametric discourse metric DiscoScore and used BERT to model discourse coherence from different perspectives driven by the central theory. All these models leverage the powerful encoding capability of BERT to fully learn deep semantic sentence information for text generation.

Pre-trained models with outstanding achievements in text generation include the generative pre-trained Transformer (GPT) series, bidirectional and
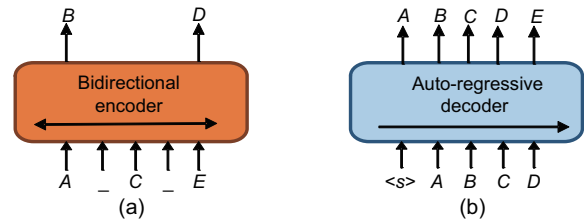

Fig. 7 Encoder-only (a) and decoder-only (b) model schematics

auto-regressive Transformers (BART) (Lewis et al., 2020), and text-to-text transfer Transformer (T5). The language learning capability possessed by pre-trained models has emerged as key support for significantly improving text generation performance. Microsoft and NVIDIA presented Megatron-Turing NLG 530B (MT-NLG), a large-scale generative language model (Smith et al., 2022). Lee and Hsiang (2020) performed fine-tuning to generate patent claims using GPT-2. Ma et al. (2022) proposed the Switch-GPT model to generate fluent and well-articulated sentences using the powerful language

modeling capabilities of GPT-2. Alexandr et al. (2021) exploited GPT-3's ability to summarize text and applied it to manually generated summaries after fine-tuning the Russian news corpus. Liu Y et al. (2021) proposed a new knowledge graph-augmented pre-trained language generation model, KG-BART, to generate logical sentences as the output by including complex relationships between concepts through knowledge graphs. Most researchers exploit large pre-training to generate models that are simply fine-tuned for a specific task or add neural network downstream of the model to achieve the goal. Therefore, exploring lightweight pre-trained models is urgently needed.

The GPT series of models introduced by OpenAI are the most successful large model for text generation. The GPT model implements the pre-trained fine-tuning framework for the first time, which allows different input and output transformations to be applied according to various tasks, thus making the downstream tasks closer to the upstream pre-trained models. The foundation of the GPT model is the Transformer model, and the optimization target is the forward language model. As shown in Fig. 7b, the model is requested to decode the output of the next predicted word through the input of the four letters A, B, C, and D. The full-text information is invisible during training, but only the prior information, which belongs to the decoder-only model. Therefore, the model is more suitable for text generation tasks, such as text summarization, paraphrasing, story generation, and copywriting generation.

As the GPT model is continuously updated, its number of parameters and range of training data are expanded, and the depth of the model architecture is increased, all of which help improve the language learning ability and generative power of the GPT model. ChatGPT, currently the most successful model in the text generation field, is based on the GPT-3.5 series fine-tuned and trained by reinforcement learning with human feedback (RLHF). Compared with other GPT models, ChatGPT has a rich corpus that enables a better grasp of the complexity of human language. In addition, ChatGPT is robustly adaptive and self-learning, is capable of fine-tuning for various scenarios and tasks, and continuously learns from constant exposure to new corpora, thus improving effectiveness and accuracy.

3.4 Hybrid text generation methods

With the emergence of new forms of text for social media and other forms of text, text generation has faced more challenges and opportunities. Researchers have been exploring hybrid approaches to text generation. These approaches typically require accounting for the special forms and linguistic text conventions, using rule-driven approaches to process different types of texts in a specific way, and then using deep learning for modeling. Such approaches provide the flexibility to address different types of powerful self-learning and self-adaptability text generation tasks. Common hybrid text generation methods include the following:

1. Hybrid approach based on rules and statistical models. This approach usually generates basic text using rules and templates first, and then modifies and adjusts the basic text using statistical models. For example, when generating autoresponse emails, the foundational structure of the response is first generated by using rules, and a hidden Markov model (Glennie et al., 2023) is later used to adjust the response to satisfy the specific needs of the users.

2. Hybrid approach based on statistical models and deep learning. The approach directly employs deep learning models to generate text. Then, statistical models are used for filtering and adjusting the text. For example, when generating news headlines, some alternative headlines can be generated first using a Transformer-based model and then a statistical model to filter and sort these headlines and select the headlines that best meet the requirements.

3. Hybrid approach based on rules and deep learning. This approach uses rules to generate the base text and adjusts and optimizes it by using deep learning models. When generating natural language dialogs, rules are available to generate the underlying text first. Then, RNNs are applied to analyze and adjust the semantic meaning to make models more consistent with natural language expression.

4. Graph-based text generation methods. This approach normally treats text as a graph structure and establishes connections between vertices using syntax and semantics. Then, using deep learning methods, suitable graph generation algorithms, such as variational self-encoders and GANs, are designed to generate the graph structure and the corresponding text (Fan et al., 2019; Zhao L et al., 2020). Using

graphs to represent text makes it possible to better capture the complex relationships between words. The drawback of the technique is that it requires more computational resources and time.

3.5 Summary analysis of methods

To analyze and explore the relationship between different methods more intuitively, Fig. 8 compares the advantages and disadvantages of the above four methods. It can be seen that the early rule-based text generation method has a wider range of applications, which can be adjusted based on the rule conditions to generate the structure of the text and language style and the rules are easy to re-adjust, but it cannot deal with a large-scale corpus, the generated text lack of diversity and flexibility, as well as a higher labor cost. To overcome such shortcomings, some researchers have proposed a statistic-based text generation method that has the ability to learn different language model styles based on a large amount of information, has strong adaptability and flexibility, and is more efficient compared to manual methods. However, when the data are insufficient or of poor quality, the performance of the generated text will be directly affected.

With the development of deep learning, the deep learning based text generation method is greatly favored, and the quality of the generated text is more guaranteed while having strong flexibility and adaptability. However, deep learning models have a high demand for training data and a long training time, requiring a large investment of time and resources. In addition, the process of deep learning is more implicit, which leads to poor interpretability. To fully combine the advantages of the above three text generation methods, some scholars have proposed a hybrid text generation method, which can ensure high flexibility and controllability, as well as guarantee the quality of the generated text. Nevertheless, data processing and algorithm selection need to be continuously adjusted according to different tasks. In summary, different methods have certain merits, and with the continuous upgrading of hardware equipment, deep learning and large model text generation methods will become the mainstream direction of research.

To compare the differences between the four methods more intuitively through experimental performance, we take PubMedQA (Jin et al., 2019), a large medical text generation dataset, as an example, and use ROUGE (Akter et al., 2022) as the evaluation metric for performance comparison. The performance results are shown in Table 4, in which maximal marginal relevance (MMR) (Lin J et al., 2010) is a statistic-based text generation method by weighing the relevance between the query and the documents and the redundancy between the documents and adopting the MMR algorithm to rank the document diversity and thus to perform text generation. Pointer-generator network (PGN) is a deep learning based text generation method, and TransformerABS, BERTSUMABS (Liu Y and Lapata, 2019b), and HSIG are pre-trained model-based text generation methods. Experimental comparison results show that different methods perform differently on the same task. In particular, the text generation method based on pre-trained models has the optimal performance, which is inextricably linked to the strong language learning ability of pre-trained models. The deep learning based method performs reasonably well on the text generation task, which can be considered when the data samples are insufficient. The rule-based text generation method and the statistic-based text generation method can be selected according to the text generation task.
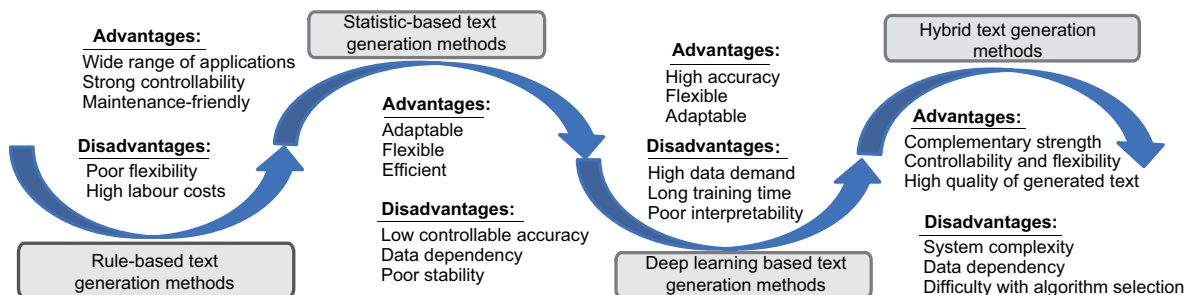

Fig. 8 Comparison of advantages and disadvantages between different methods

4 Typical applications of text generation

Text generation technology is rapidly penetrating all aspects and silently changing people's work and lives. It creates tremendous business value and continues to drive economic development and innovative industries forward. In this section, we present several typical applications of text generation technology in the industry.

4.1 Text generation in the news field

Text generation has a wide range of applications in journalism. In the past, news editors had to quickly review many news articles and select the most essential and relevant content for coverage. Given the time and human cost constraints, editors required an automated approach to process large volumes of text, which leads to frequent adoption of text summarization techniques (Lai et al., 2022).

The applications for text generation in journalism are classified mainly into keyword (Jayasiriwardene and Ganegoda, 2020), headline (Feng and Lapata, 2013), and summary (Gupta et al., 2022). These

summaries contain a wealth of key information that facilitates rapid journalistic development. In addition, writing scripts in the news requires a great deal of manpower and time, and many news organizations are now employing AI to generate scripts, especially for structured news stories, such as sports, weather, stock market trading changes, and company earnings reports. Currently, there are several relatively famous and established systems in journalism that automatically generate press releases, such as GPT-3 (Floridi and Chiriatti, 2020), AI Writer, WordSmith, Quill, and Heliograf. Table 5 summarizes some relatively mature systems, which are mostly applied to news content generation. These automated text generation systems use natural language processing and machine learning technologies to enable news organizations to report news quickly, accurately, and effectively.

4.2 Text generation in literature

Automatic document generation plays an essential role in libraries. Similar to journalism, libraries require the generation of briefs for many literature books, and manual abstracting is inefficient, so automatic abstract generation technology is needed to replace manual abstracting tasks efficiently (Doush et al., 2017). Automatic text summarization has the following main applications in library science:

1. Literature abstract generation. Libraries collect a large quantity of literature, such as journals and reports. The abstracts in the materials are typically long and time-consuming to read. Automatic literature abstract technology allows for condensing

Table 4 Performance comparison of different methods on PubMedQA dataset

Method	ROUGE-1	ROUGE-2	ROUGE-L
Lead3	30.86	9.77	21.15
MMR	30.11	9.02	24.39
TransformerABS	29.65	9.30	26.63
PGN	32.91	11.54	28.11
BERTSUMABS	35.29	13.63	30.71
HSIG	37.98	15.51	32.41

Table 5 News-related system

Application	R&D unit	Introduction
GPT-3	OpenAI	The product generates natural and fluent language, and it is capable of engaging in dialogue, translation, summary, Q&A, and many other tasks. Some news organizations are already experimenting with GPT-3 to automatically generate news releases
AI Writer	36Kr	The application automatically generates press releases, reviews, stories, and other content. AI Writer is characterized by speed, efficiency, and accuracy
WordSmith	Automated Insights	The system works to generate content for press releases, financial report analysis, etc. The system automatically analyzes data and generates natural language text abstracts
Quill	Automated Insights	The application is designed to generate text for sports, business, finance, and other fields. The system works with natural language generation techniques to generate high-quality text content
Heliograf	Washington Post	The software automates the generation of simple news releases, such as event reports and election results

abstracts into a collection of critical information, making it easier for readers to understand the content (Dowling and Lucey, 2023).

2. Summary generation of bibliographic information. In an online public library catalog, the description of each book contains a large amount of information, such as title, author, publisher, publication date, and ISBN. Automatic text summarization technology allows the information to be compressed into concise and accurate descriptions to facilitate readers finding the desired book more quickly (Stevens, 2022).

3. Library service introduction summary generation. Libraries provide many services, such as borrowing, returning, and consulting, and each service has a detailed introduction. Automatic text summarization technology enables us to compress this introductory information into a brief and precise description to help readers better understand and access the services (Lund and Wang, 2023).

4. Reader comment summary generation. Library websites and applications allow readers to post comments and ratings. These reviews are typically long, and they require considerable staff time to understand readers' thoughts. Automatic text summarization technology enables these reviews to be summarized into short, accurate summaries to help readers better understand what other readers think of the book (Barker et al., 2016). These application directions help libraries improve service quality and enhance reading experience while reducing the workload of library staff with improved efficiency.

Many sophisticated automated text generation systems already exist in the book and literature fields. The LibrarianAI system employs natural language processing technology to automatically generate book classifications, indexes, and bibliographic descriptions to improve library efficiency and service quality. The SummarizeBot application uses AI technology to generate summaries of various texts, such as emails, news articles, papers, and reports. The library leverages SummarizeBot to generate literature summaries to assist patrons in understanding the content of the literature more directly (Karakoylu et al., 2020). In addition, text generation in the book field can be content continuation based on the provided text, including literary creation, business writing, and teaching content. Typical applications are Wordcraft launched

by Google (Coenen et al., 2021) and GPT-3 launched by OpenAI (Floridi and Chiriatti, 2020). Wordcraft was launched in 2021 by Google, which allows authors to dialogue with AI to write stories, supporting for continuation, expansion, rewriting, generating outlines, and other features that greatly improve writing efficiency.

4.3 Text generation in the retrieval field

Text generation has wide applications in the retrieval field. Retrieval systems are designed to find information of interest to users from considerable text data. However, finding the required information in massive data is time-consuming and labor-intensive. Using text generation allows users to quickly understand the information and enables retrieval systems to locate relevant information quickly (Cai et al., 2022). Some text generation applications in the field of retrieval are as follows:

1. Summary generation. The search system allows automatic text generation to extract and present important information to the user. These summaries are displayed in the search result list, helping users quickly comprehend the document content and thus improve the relevant document selection.

2. Text inquiry. The user is allowed to enter a query, and the system then provides the most relevant text, which helps the user quickly understand the query results and determine whether to view the document further.

3. Text clustering. Documents are classified into different groups so that the user can find relevant information in a collection of similar documents. Text generation assists in identifying similarities between documents, so the documents can be grouped together.

4. Text categorization. Documents are classified into different categories so that users can find relevant information. The summary serves to determine the category to which the document belongs and thus categorizes it correctly.

4.4 Text generation in other fields

In addition to the application areas mentioned above, text generation techniques are used in a wide range of other fields.

1. In academic research, text generation helps

researchers quickly understand the main content and conclusions of a paper to determine whether it corresponds to their research interests.

2. In business analysis, text generation enables companies to quickly understand key information such as the marketing strategies and product features of competitors.

3. In legal documents, text generation enables lawyers to quickly understand the main content and key facts of a case, thus allowing lawyers to better serve clients.

4. In medical research, text generation helps researchers quickly understand a large quantity of medical literature to determine the focus and direction of their research.

5. In the movie field, a brief overview of the movie plot and content is provided to generate a movie synopsis, which quickly assists viewers in understanding the movie content, improving the movie-going experience and promoting movie marketing.

Table 6 summarizes several other representative text generation products and the corresponding product features. Text generation is now beginning to be explored in the life sciences, medical, manufacturing, materials science, media, entertainment, automotive, and aerospace fields. Regardless of the field, text generation technology helps users deliver creative work on demand, thus bringing tremendous productivity gains to each field.

5 Challenge analysis

Although AI text generation has demonstrated powerful capabilities with continuous development of deep learning and pre-trained models, it still faces many challenges and difficulties.

1. Interpretability problem. The black-box nature of deep learning models makes it difficult to interpret the generated texts, which is not conducive to error analysis or targeted solutions (Rudin, 2019).

The quality of the text generated by the model is difficult to guarantee, and its credibility is reduced, thus affecting its applications.

2. Knowledge acquisition and update problem. Language models require a large quantity of text data for training, which has to be normalized and standardized. It is necessary to ensure that the data are constantly updated to facilitate better learning of machine learning algorithms. Processing requires many human and financial resources. Therefore, it is critical to automate the process of obtaining accurate, diverse, and representative data.

3. Text quality problem. Text quality is reflected mainly in the following: First, the generated text requires fluency. Text generation requires natural fluency and accurate expression of the true input semantics. Most text generation copies and adapts only the original data, not truly digging deeper into their semantic relationships, which often leads to some overfitting problems and a lack of certain logical reasoning abilities. Second, the generated text needs to maintain authenticity. Although the content of the text generation model is logically correct overall, it suffers from the problem of violating common sense or facts. Third, the generated text has an unknown provenance. The generated content comes from fusing multiple sources, and the provenance cannot be given explicitly.

4. Automatic text evaluation problem. Humans can judge the quality of text through subjective awareness, but it is difficult for computers to identify an automatic evaluation index that accounts for the quality of the text for an objective and accurate assessment (van der Lee et al., 2019). The currently employed evaluation metrics based on the degree of N-gram overlap, e.g., BLEU (Evtikhiev et al., 2023) and ROUGE (Akter et al., 2022), and model-driven evaluation metrics (Zhang TY et al., 2020) have difficulty in reliably evaluating text quality. These evaluation criteria lead to difficulties in accurately

Table 6 Representative text generation products

Application	R&D unit	Product feature
ChatGPT	OpenAI	Text generation, summarizing, and answering questions
InstructGPT	OpenAI	Text generation, summarizing, and answering questions
Jasper	Jasper.ai	Writing blog posts, social media posts, and website copy
Copi.ai	Copi.ai	Text generation, chatbots, sentiment analysis, and customer support automation
CopySmith	CopySmith	Creating marketing copy, product descriptions, and advertising copy
Boost Sport AI	ARRIA	Performance analysis, creation of advertising copy, and dialogues

defining and formalizing the optimization goals of text generation tasks, which is a non-negligible deficiency of existing text generation models.

5. Model capital consumption problem. Many current pre-trained language models have complex architectures and significant hardware equipment requirements, and need considerable energy and capital investments. A large amount of computing power also leads to an increase in carbon emissions, which poses a considerable challenge to environmental protection and climate change. Lightweight text generation models will have much broader prospects.

6. Copyright dispute problem. AI generates highly similar or even identical content, which leads to disputes with the original creator or copyright holder. Simultaneously, there is a risk that works generated by AI are infringed upon by others (Selvadurai and Matulionyte, 2020). Therefore, it is crucial to reduce the possibility of copyright disputes by determining how to induce diversity in the generated content or automatically mark the provenance of the generated similar content to ensure that the generated content is safe, legal, and ethical.

6 Prospective trends and conclusions

With the increasing maturity and reliability of AI technology, machines can realize truly intelligent dialog and communication through deep learning algorithms and the natural language processing technology. As a new communication method, AI text generation provides users with more humanized and intelligent services and support, which improves user satisfaction and experience. The future development trend of text generation includes mainly the following directions:

1. Enhanced quality and quantity of text generation. As machine learning and natural language processing technologies continue to advance, text generation models will become more intelligent and efficient. These will be capable of understanding human language better and generating substantially more and better text within time and budget constraints.

2. Social and interactive text generation. The future of text generation will focus more on social and interactive aspects. Text generation enables the generation of more interactive text, allowing more natural interactions between machines and humans.

3. Data-based natural language generation.

The future of text generation will be more data-dependent. Using big data analysis, future text generation will better understand the semantics and usage norms of human language to generate more accurate and logically structured texts.

4. Multilingual text generation. The future of text generation will be more adaptable to multiple languages and cultures. Machine learning and natural language processing techniques are expected to better understand the differences between languages and build cross-linguistic and cross-cultural text generation models.

5. Adaptive text generation. The future of text generation will be more human and intelligent. Machine learning and natural language processing technologies will be better able to generate more personalized and adaptive text through natural human language interaction and contextual understanding.

From a comprehensive perspective, text generation will tend to develop rapidly in the directions of automation, personalization, multilingualism, real time, and style diversification, thus meeting different needs. The emergence of ChatGPT means that in the future, AI technology will gradually replace some repetitive and inefficient human work, increasing efficiency and lowering costs for enterprises while providing higher-level career development for employees. Finally, the emergence of ChatGPT creates some issues and challenges, such as social and ethical issues arising from the replacement of data privacy and human intelligence, which requires us to always be vigilant and aware when using ChatGPT.

In this paper, we provide an overview of the main forms of text generation and some recent research advances. We also present an objective analysis of the application areas and challenges faced by text generation, and finally propose an overview of several future research trends. We believe that intelligent text generation will become an essential area for the future development of multimodal AI.

Contributors

Bing LI designed the research and drafted the paper. Peng YANG oversaw and led the planning and execution of the study. Yuankang SUN, Zhongjian HU, and Meng YI collected the information and revised the paper.

Compliance with ethics guidelines

All the authors declare that they have no conflict of interest.

References

Adamopoulou E, Moussiades L, 2020. Chatbots: history, technology, and applications. *Mach Learn Appl*, 2:100006. https://doi.org/10.1016/j.mlwa.2020.100006

Akter M, Bansal N, Karmaker SK, 2022. Revisiting automatic evaluation of extractive summarization task: can we do better than ROUGE? Findings of the Association for Computational Linguistics, p.1547-1560. https://doi.org/10.18653/v1/2022.findings-acl.122

Albawi S, Mohammed TA, Al-Zawi S, 2017. Understanding of a convolutional neural network. Proc Int Conf on Engineering and Technology, p.1-6. https://doi.org/10.1109/ICEngTechnol.2017.8308186

Alexandr N, Irina O, Tatyana K, et al., 2021. Fine-tuning GPT-3 for Russian text summarization. Proc 5th Computational Methods in Systems and Software, p.748-757. https://doi.org/10.1007/978-3-030-90321-3_61

Barker E, Paramita ML, Aker A, et al., 2016. The SENSEI annotated corpus: human summaries of reader comment conversations in on-line news. Proc 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, p.42-52. https://doi.org/10.18653/v1/W16-3605

Barrera A, Verma R, 2012. Combining syntax and semantics for automatic extractive single-document summarization. Proc 13th Int Conf on Computational Linguistics and Intelligent Text Processing, p.366-377. https://doi.org/10.1007/978-3-642-28601-8_31

Birjali M, Kasri M, Beni-Hssane A, 2021. A comprehensive survey on sentiment analysis: approaches, challenges and trends. *Knowl-Based Syst*, 226:107134. https://doi.org/10.1016/j.knosys.2021.107134

Cai D, Wang Y, Liu L, et al., 2022. Recent advances in retrieval-augmented text generation. Proc 45th Int ACM SIGIR Conf on Research and Development in Information Retrieval, p.3417-3419. https://doi.org/10.1145/3477495.3532682

Cao JR, Wang CW, 2018. Social media text generation based on neural network model. Proc 2nd Int Conf on Computer Science and Artificial Intelligence, p.58-61. https://doi.org/10.1145/3297156.3297206

Chen YC, Gan Z, Cheng Y, et al., 2020. Distilling knowledge learned in BERT for text generation. Proc 58th Annual Meeting of the Association for Computational Linguistics, p.7893-7905. https://doi.org/10.18653/v1/2020.acl-main.705

Chen ZH, Shen YL, Song Y, et al., 2021. Cross-modal memory networks for radiology report generation. Proc 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing, p.5904-5914. https://doi.org/10.18653/v1/2021.acl-long.459

Christian H, Agus MP, Suhartono D, 2016. Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF). *ComTech*, 7(4):285-294. https://doi.org/10.21512/comtech.v7i4.3746

Coenen A, Davis L, Ippolito D, et al., 2021. Wordcraft: a human-AI collaborative editor for story writing. https://doi.org/10.48550/arXiv.2107.07430

Croft R, Newlands D, Chen ZY, et al., 2021. An empirical study of rule-based and learning-based approaches for static application security testing. Proc 15th ACM/IEEE Int Symp on Empirical Software Engineering and Measurement, Article 8. https://doi.org/10.1145/3475716.3475781

Dathathri S, Madotto A, Lan J, et al., 2020. Plug and play language models: a simple approach to controlled text generation. Proc 8th Int Conf on Learning Representations.

Devlin KJ, Chang MW, Lee K, 2019. BERT: pre-training of deep bidirectional Transformers for language understanding. Proc Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p.4171-4186. https://doi.org/10.18653/v1/N19-1423

Dey R, Salem FM, 2017. Gate-variants of gated recurrent unit (GRU) neural networks. Proc IEEE 60th Int Midwest Symp on Circuits and Systems, p.1597-1600. https://doi.org/10.1109/MWSCAS.2017.8053243

Ding HJ, Xu XL, 2023. SAN-T2T: an automated table-to-text generator based on selective attention network. *Nat Lang Eng*, First View. https://doi.org/10.1017/S135132492300013X

Doush IA, Alkhateeb F, Albsoul A, 2017. AraDaisy: a system for automatic generation of ARABIC daisy books. *Int J Comput Appl Technol*, 55(4):322-333. https://doi.org/10.1504/IJCAT.2017.086007

Dowling M, Lucey B, 2023. ChatGPT for (finance) research: the bananarama conjecture. *Financ Res Lett*, 53:103662. https://doi.org/10.1016/j.frl.2023.103662

Evtikhiev M, Bogomolov E, Sokolov Y, et al., 2023. Out of the BLEU: how should we assess quality of the code generation models? *J Syst Softw*, 203:111741. https://doi.org/10.1016/j.jss.2023.111741

Fan A, Gardent C, Braud C, et al., 2019. Using local knowledge graph construction to scale Seq2Seq models to multi-document inputs. Proc Conf on Empirical Methods in Natural Language Processing and the 9th Int Joint Conf on Natural Language Processing, p.4186-4196. https://doi.org/10.18653/v1/D19-1428

Fei H, Li CL, Ji DH, et al., 2022. Mutual disentanglement learning for joint fine-grained sentiment classification and controllable text generation. Proc 45th Int ACM SIGIR Conf on Research and Development in Information Retrieval, p.1555-1565. https://doi.org/10.1145/3477495.3532029

Feng YS, Lapata M, 2013. Automatic caption generation for news images. *IEEE Trans Patt Anal Mach Intell*, 35(4):797-812. https://doi.org/10.1109/TPAMI.2012.118

Floridi L, Chiriatti M, 2020. GPT-3: its nature, scope, limits, and consequences. *Minds Mach*, 30(4):681-694. https://doi.org/10.1007/s11023-020-09548-1

Gao S, Ren ZC, Zhao YH, et al., 2019. Product-aware answer generation in e-commerce question-answering. Proc 12th ACM Int Conf on Web Search and Data Mining, p.429-437. https://doi.org/10.1145/3289600.3290992

Garbacea C, Mei QZ, 2020. Neural language generation: formulation, methods, and evaluation. https://doi.org/10.48550/arXiv.2007.15780

Glennie R, Adam T, Leos-Barajas V, et al., 2023. Hidden Markov models: pitfalls and opportunities in ecology. *Methods Ecol Evol*, 14(1):43-56. https://doi.org/10.1111/2041-210X.13801

Goodfellow I, Pouget-Abadie J, Mirza M, et al., 2020. Generative adversarial networks. *Commun ACM*, 63(11):139-144. https://doi.org/10.1145/3422622

Guo ZX, Yan MX, Qi JX, et al., 2023. Few-shot table-to-text generation with prompt planning and knowledge memorization. https://doi.org/10.48550/arXiv.2302.04415

Gupta A, Chugh D, Anjum, 2022. Automated news summarization using transformers. Int Conf on Sustainable Advanced Computing, p.249-259. https://doi.org/10.1007/978-981-16-9012-9_21

He XL, Xu QK, Zeng Y, et al., 2022. CATER: intellectual property protection on text generation APIs via conditional watermarks. Proc 36th Conf on Neural Information Processing Systems.

Hua XY, Wang L, 2020. PAIR: planning and iterative refinement in pre-trained Transformers for long text generation. Proc Conf on Empirical Methods in Natural Language Processing, p.781-793. https://doi.org/10.18653/v1/2020.emnlp-main.57

Imam AT, Rousan T, Aljawarneh S, 2014. An expert code generator using rule-based and frames knowledge representation techniques. Proc 5th Int Conf on Information and Communication Systems, p.1-6. https://doi.org/10.1109/IACS.2014.6841951

Jayasiriwardene TD, Ganegoda GU, 2020. Keyword extraction from tweets using NLP tools for collecting relevant news. Proc Int Research Conf on Smart Computing and Systems Engineering, p.129-135. https://doi.org/10.1109/SCSE49731.2020.9313024

Jin Q, Dhingra B, Liu ZP, et al., 2019. PubMedQA: a dataset for biomedical research question answering. Proc 9th Int Joint Conf on Natural Language Processing, p.2567-2577. https://doi.org/10.18653/v1/D19-1259

Jungwirth D, Haluza D, 2023. Artificial intelligence and the sustainable development goals: an exploratory study in the context of the society domain. *J Softw Eng Appl*, 16(4):91-112. https://doi.org/10.4236/jsea.2023.164006

Karakoylu PE, Kural AG, Gulden S, 2020. The effect of artificial intelligence (AI) on new product development (NPD): a future scenario. *IOP Conf Ser Mater Sci Eng*, 960:022026. https://doi.org/10.1088/1757-899X/960/2/022026

Keskar NS, McCann B, Varshney LR, et al., 2019. CTRL: a conditional transformer language model for controllable generation. https://doi.org/10.48550/arXiv.1909.05858

King MR, 2023. A place for large language models in scientific publishing, apart from credited authorship. *Cell Mol Bioeng*, 16(2):95-98. https://doi.org/10.1007/s12195-023-00765-z

King MR, ChatGPT, 2023. A conversation on artificial intelligence, chatbots, and plagiarism in higher education. *Cell Mol Bioeng*, 16(1):1-2. https://doi.org/10.1007/s12195-022-00754-8

Kingma DP, Welling M, 2014. Auto-encoding variational Bayes. https://arxiv.org/abs/1312.6114

Koncel-Kedziorski R, Bekal D, Luan Y, et al., 2019. Text generation from knowledge graphs with graph transformers. Proc Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p.2284-2293. https://doi.org/10.18653/v1/N19-1238

Koupaee M, Wang WY, 2018. WikiHow: a large scale text summarization dataset. https://doi.org/10.48550/arXiv.1810.09305

Kraus S, Castellanos I, Albermann M, et al., 2016. Using Arden syntax for the generation of intelligent intensive care discharge letters. *Stud Health Technol Inform*, 228:471-475. https://doi.org/10.3233/978-1-61499-678-1-471

Lai V, Smith-Renner A, Zhang K, et al., 2022. An exploration of post-editing effectiveness in text summarization. Proc Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p.475-493. https://doi.org/10.18653/v1/2022.naacl-main.35

Lee JS, Hsiang J, 2020. Patent claim generation by fine-tuning OpenAI GPT-2. *World Patent Inform*, 62:101983. https://doi.org/10.1016/j.wpi.2020.101983

Leppänen L, Munezero M, Granroth-Wilding M, et al., 2017. Data-driven news generation for automated journalism. Proc 10th Int Conf on Natural Language Generation, p.188-197. https://doi.org/10.18653/v1/W17-3528

Lewis M, Liu YH, Goyal N, et al., 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. Proc 58th Annual Meeting of the Association for Computational Linguistics, p.7871-7880. https://doi.org/10.18653/v1/2020.acl-main.703

Li B, Yang P, Zhao HL, et al., 2023. Hierarchical sliding inference generator for question-driven abstractive answer summarization. *ACM Trans Inform Syst*, 41(1):7. https://doi.org/10.1145/3511891

Li JJ, Song HZ, Li J, 2022. Transformer-based question text generation in the learning system. Proc 6th Int Conf on Innovation in Artificial Intelligence, p.50-56. https://doi.org/10.1145/3529466.3529484

Li YJ, Choi D, Chung J, et al., 2022. Competition-level code generation with AlphaCode. *Science*, 378(6624):1092-1097. https://doi.org/10.1126/science.abq1158

Liang ZY, Du JP, Shao YX, et al., 2021. Gated graph neural attention networks for abstractive summarization. *Neurocomputing*, 431:128-136. https://doi.org/10.1016/j.neucom.2020.09.066

Lin J, Madnani N, Dorr BJ, 2010. Putting the user in the loop: interactive maximal marginal relevance for query-focused summarization. Proc Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p.305-308.

Lin JY, Sun X, Ma SM, et al., 2018. Global encoding for abstractive summarization. Proc 56th Annual Meeting of the Association for Computational Linguistics, p.163-169. https://doi.org/10.18653/v1/P18-2027

Liu TY, Wang KX, Sha L, et al., 2018. Table-to-text generation by structure-aware Seq2Seq learning. Proc 32nd AAAI Conf on Artificial Intelligencee, 32(1). https://doi.org/10.1609/aaai.v32i1.11925

Liu Y, Lapata M, 2019a. Hierarchical Transformers for multi-document summarization. Proc 57th Annual Meeting of the Association for Computational Linguistics, p.5070-5081. https://doi.org/10.18653/v1/P19-1500

Liu Y, Lapata M, 2019b. Text summarization with pretrained encoders. Proc Conf on Empirical Methods in Natural Language Processing and the 9th Int Joint Conf on Natural Language Processing, p.3730-3740. https://doi.org/10.18653/v1/D19-1387

Liu Y, Wan Y, He LF, et al., 2021. KG-BART: knowledge graph-augmented BART for generative commonsense reasoning. Proc 35th AAAI Conf on Artificial Intelligence, p.6418-6425. https://doi.org/10.1609/aaai.v35i7.16796

Liu YH, Ott M, Goyal N, et al., 2019. RoBERTa: a robustly optimized BERT pretraining approach. https://doi.org/10.48550/arXiv.1907.11692

Lund BD, Wang T, 2023. Chatting about ChatGPT: how may AI and GPT impact academia and libraries? Lib Hi Tech News, 40(3):26-29. https://doi.org/10.1108/LHTN-01-2023-0009

Luo Y, Zhao SL, Li XC, et al., 2016. Text keyword extraction method based on word frequency statistics. J Comput Appl, 36(3):718-725 (in Chinese). https://doi.org/10.11772/j.issn.1001-9081.2016.03.718

Ma C, Zhang S, Shen GH, et al., 2022. Switch-GPT: an effective method for constrained text generation under few-shot settings (student abstract). Proc 36th AAAI Conf on Artificial Intelligence, p.13011-13012. https://doi.org/10.1609/aaai.v36i11.21642

Meng R, Zhao SQ, Han SG, et al., 2017. Deep keyphrase generation. Proc 55th Annual Meeting of the Association for Computational Linguistics, p.582-592. https://doi.org/10.18653/v1/P17-1054

Miller D, 2019. Leveraging BERT for extractive text summarization on lectures. https://doi.org/10.48550/arXiv.1906.04165

Mukherjee S, 2021. Sentiment analysis. In: Mukherjee S (Ed.), MLNET Revealed: Simple Tools for Applying Machine Learning to Your Applications. Apress, Berkeley, p.113-127. https://doi.org/10.1007/978-1-4842-6543-7_7

Oh C, Choi J, Lee S, et al., 2020. Understanding user perception of automated news generation system. Proc CHI Conf on Human Factors in Computing Systems, p.1-13. https://doi.org/10.1145/3313831.3376811

Omar R, Mangukiya O, Kalnis P, et al., 2023. ChatGPT versus traditional question answering for knowledge graphs: current status and future directions towards knowledge graph chatbots. https://doi.org/10.48550/arXiv.2302.06466

Özateş ŞB, Özgür A, Radev D, 2016. Sentence similarity based on dependency tree kernels for multi-document summarization. Proc 10th Int Conf on Language Resources and Evaluation, p.2833-2838.

Palangi H, Deng L, Shen YL, et al., 2016. Deep sentence embedding using long short-term memory networks: analysis and application to information retrieval. IEEE/ACM Trans Audio Speech Lang Process, 24(4):694-707. https://doi.org/10.1109/TASLP.2016.2520371

Procter R, Arana-Catania M, He YL, et al., 2023. Some observations on fact-checking work with implications for computational support. https://doi.org/10.48550/arXiv.2305.02224

Rudin C, 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat Mach Intell, 1(5):206-215. https://doi.org/10.1038/s42256-019-0048-x

Saeed A, Ilić S, Zangerle E, 2019. Creative GANs for generating poems, lyrics, and metaphors. https://doi.org/10.48550/arXiv.1909.09534

See A, Liu PJ, Manning CD, 2017. Get to the point: summarization with pointer-generator networks. Proc 55th Annual Meeting of the Association for Computational Linguistics, p.1073-1083. https://doi.org/10.18653/v1/P17-1099

Selvadurai N, Matulionyte R, 2020. Reconsidering creativity: copyright protection for works generated using artificial intelligence. J Intellect Prop Law Pract, 15(7):536-543. https://doi.org/10.1093/jiplp/jpaa062

Senftleben M, Buijtelaar L, et al., 2020. Robot creativity: an incentive-based neighbouring rights approach. Eur Intellect Prop Rev, 42(12):717. https://doi.org/10.2139/ssrn.3707741

Smith S, Patwary M, Norick B, et al., 2022. Using DeepSpeed and megatron to train Megatron-Turing NLG 530B, a large-scale generative language model. https://doi.org/10.48550/arXiv.2201.11990

Stevens K, 2022. Retrospective enhancement of bibliographic records with tables of contents and summary data and the effect on collection usage. J Aust Lib Inform Assoc, 71(4):379-387. https://doi.org/10.1080/24750158.2022.2135359

Sun XF, Meng YX, Ao X, et al., 2022. Sentence similarity based on contexts. Trans Assoc Comput Ling, 10:573-588. https://doi.org/10.1162/tacl_a_00477

Surameery NMS, Shakor MY, 2023. Use ChatGPT to solve programming bugs. Int J Inform Technol Comput Eng, 3(1):17-22. https://doi.org/10.55529/ijitc.31.17.22

Svyatkovskiy A, Deng SK, Fu SY, et al., 2020. IntelliCode compose: code generation using transformer. Proc 28th ACM Joint Meeting on European Software Engineering Conf and Symp on the Foundations of Software Engineering, p.1433-1443. https://doi.org/10.1145/3368089.3417058

Thoppilan R, De Freitas D, Hall J, et al., 2022. LaMDA: language models for dialog applications. https://doi.org/10.48550/arXiv.2201.08239

Tian YF, Peng NY, 2022. Zero-shot sonnet generation with discourse-level planning and aesthetics features. Proc Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p.3587-3597. https://doi.org/10.18653/v1/2022.naacl-main.262

Tomczak JM, Welling M, 2016. Improving variational autoencoders using householder flow. https://doi.org/10.48550/arXiv.1611.09630

Tran OT, Bui VT, 2021. Neural text normalization in speech-to-text systems with rich features. Appl Artif Intell, 35(3):193-205. https://doi.org/10.1080/08839514.2020.1842108

Trivedi A, Pant N, Shah P, et al., 2018. Speech to text and text to speech recognition systems—a review. *IOSR J Comput Eng*, 20(2):36-43. https://doi.org/10.9790/0661-2002013643

van der Lee C, Gatt A, van Miltenburg E, et al., 2019. Best practices for the human evaluation of automatically generated text. Proc 12th Int Conf on Natural Language Generation, p.355-368. https://doi.org/10.18653/v1/W19-8643

Vaswani A, Shazeer N, Parmar N, et al., 2017. Attention is all you need. Proc 31st Int Conf on Neural Information Processing Systems, p.6000-6010. https://doi.org/10.5555/3295222.3295349

Vodolazova T, Lloret E, 2019. The impact of rule-based text generation on the quality of abstractive summaries. Proc Int Conf on Recent Advances in Natural Language Processing, p.1275-1284. https://doi.org/10.26615/978-954-452-056-4_146

Wang CH, Tang Y, Ma XT, et al., 2020. FAIRSEQ S2T: fast speech-to-text modeling with FAIRSEQ. Proc 1st Conf of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th Int Joint Conf on Natural Language Processing: System Demonstrations, p.33-39.

Wang TM, Wan XQ, Jin HQ, 2020. AMR-to-text generation with graph Transformer. *Trans Assoc Comput Ling*, 8:19-33. https://doi.org/10.1162/tacl_a_00297

Wang WM, Li Z, Wang JW, et al., 2017. How far we can go with extractive text summarization? Heuristic methods to obtain near upper bounds. *Exp Syst Appl*, 90:439-463. https://doi.org/10.1016/j.eswa.2017.08.040

Xu HH, Savelka J, Ashley KD, 2021. Accounting for Sentence Position and Legal Domain Sentence Embedding in Learning to Classify Case Sentences. IOS Press, Vilnius, Lithuania, p.33-42.

Yellin DM, Weiss G, 2021. Synthesizing context-free grammars from recurrent neural networks. Proc 27th Int Conf on Tools and Algorithms for the Construction and Analysis of Systems, p.351-369. https://doi.org/10.1007/978-3-030-72016-2_19

Yu X, Vu NT, Kuhn J, 2019. Learning the dyck language with attention-based Seq2Seq models. Proc ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, p.138-146. https://doi.org/10.18653/v1/W19-4815

Yu Y, Si XS, Hu CH, et al., 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neur Comput*, 31(7):1235-1270. https://doi.org/10.1162/neco_a_01199

Zhang TY, Kishore V, Wu F, et al., 2020. BERTScore: evaluating text generation with BERT. Proc 8th Int Conf on Learning Representations.

Zhang XY, Zou YY, Zhang HN, et al., 2022a. Automatic product copywriting for e-commerce. Proc 36th AAAI Conf on Artificial Intelligence, p.12423-12431. https://doi.org/10.1609/aaai.v36i11.21508

Zhang XY, Shen K, Zhang C, et al., 2022b. Scenario-based multi-product advertising copywriting generation for e-commerce. https://doi.org/10.48550/arXiv.2205.10530

Zhang YQ, Huang ML, 2019. Overview of the NTCIR-14 short text generation subtask: emotion generation challenge. Proc 14th NTCIR Conf on Evaluation of Information Access Technologies, p.316-327.

Zhang ZY, Han X, Liu ZY, et al., 2019. ERNIE: enhanced language representation with informative entities. Proc 57th Annual Meeting of the Association for Computational Linguistics, p.1441-1451. https://doi.org/10.18653/v1/P19-1139

Zhao GZ, Yang P, 2022. Table-based fact verification with self-labeled keypoint alignment. Proc 29th Int Conf on Computational Linguistics, p.1401-1411.

Zhao L, Xu JJ, Lin JY, et al., 2020. Graph-based multi-hop reasoning for long text generation. https://doi.org/10.48550/arXiv.2009.13282

Zhao W, Peyrard M, Liu F, et al., 2019. MoverScore: text generation evaluating with contextualized embeddings and earth mover distance. Proc Conf on Empirical Methods in Natural Language Processing and the 9th Int Joint Conf on Natural Language Processing, p.563-578. https://doi.org/10.18653/v1/D19-1053

Zhao W, Strube M, Eger S, 2022. DiscoScore: evaluating text generation with BERT and discourse coherence. Proc 17th Conf of the European Chapter of the Association for Computational Linguistics, p.3865-3883. https://doi.org/10.18653/v1/2023.eacl-main.278

Zheng LX, Ma S, Chen ZX, et al., 2021. Ensuring the correctness of regular expressions: a review. *Int J Autom Comput*, 18(4):521-535. https://doi.org/10.1007/s11633-021-1301-4

Zhu CG, Yang ZY, Gmyr R, et al., 2021. Leveraging lead bias for zero-shot abstractive news summarization. Proc 44th Int ACM SIGIR Conf on Research and Development in Information Retrieval, p.1462-1471. https://doi.org/10.1145/3404835.3462846

Zhu QH, Luo JX, 2023. Generative design ideation: a natural language generation approach. In: Gero JS (Ed.), Design Computing and Cognition'22. Springer, Cham, p.39-50. https://doi.org/10.1007/978-3-031-20418-0_3

Zogopoulos V, Geurts E, Gors D, et al., 2022. Authoring tool for automatic generation of augmented reality instruction sequence for manual operations. *Procedia CIRP*, 106:84-89. https://doi.org/10.1016/j.procir.2022.02.159