Academic Journal of Natural Science
Journal Home: http://ajns.suaspress.org | CODEN: AJNSAE
Vol. 2, No. 2, 2025 | ISSN 3078-5170 (Print) | ISSN 3078-5189 (Online) | ISSN 3078-5197 (Digital)

A Technical Review of Sequence-to-Sequence Models

BO, Tao [1*] LI, Weiyi [2] LIU, Yue [1]

[1] TikTok.Inc, USA
[2] Georgia Institute of Technology, USA

**** BO, Tao is the corresponding author, E-mail: taobo1996@outlook.com***

Abstract: Seq2Seq models and their variants have become a mainstay of modern natural language processing and sequence modelling tasks. Just Information about Seq2Seq models. In this paper, we provide a comprehensive overview of the evolution of Seq2Seq architecture from early-stage RNN based approaches to recent Transformer based methods. The paper extensively covers additional important methods such as attention mechanisms, bidirectional encoders, pointer-generator networks, as well as optimization methods such as beam search, scheduled sampling and reinforcement learning. It also discusses the challenges of data preprocessing, loss functions, and evaluation metrics, as well as applications in machine translation, summarization, speech recognition, and conversational AI. This paper provides a comprehensive report on the design and future directions of Seq2Seq models emphasizing on theoretical foundations as well as real world applications.

Keywords: Sequence-to-sequence, Transformer, Attention Mechanism, Neural Machine Translation, Text Summarization, Conversational AI, Reinforcement Learning, Pointer-generator Network, Beam Search, Natural Language Processing.

Disciplines: Computer Science. **Subjects:** Artificial Intelligence.

DOI: https://doi.org/10.70393/616a6e73.323834 **ARK:** https://n2t.net/ark:/40704/AJNS.v2n2a01

1 INTRODUCTION

Sequence-to-sequence (Seq2Seq) models are a class of deep learning architectures designed for sequence transformation tasks, where an input sequence is mapped to an output sequence of varying length. These models have been widely adopted in applications such as machine translation, speech recognition, text summarization, and conversational AI. By leveraging an encoder-decoder structure, Seq2Seq models efficiently process sequential data and capture long-range dependencies, making them highly flexible and effective for a broad range of tasks.

The Seq2Seq models will be important in learning meaningful sequences as it is able to capture relations between tokens over long distances. While originally presented as recurrent neural networks (RNNs) or their variants like long short-term memory (LSTM) and gated recurrent units (GRU), Seq2Seq models have seen considerable evolution. Attention mechanism eliminating issues of Long-Range Dependencies and improved performance focusing on specific parts of the input during decoding (Bahdanau et al. Subsequent advancements, such as the Transformer architecture that replaced subnetworks with self-attention mechanisms, have pushed these Seq2Seq models to further revolutionize NLP and sequence modeling tasks with state-of-the-art results.1[1]

This review provides an in-depth analysis of the foundations of Seq2Seq models, their architectural advancements, and key enhancements such as attention, Transformer networks, and pre-trained language models (e.g., BERT, T5, GPT). It also explores training methodologies, real-world applications, and ongoing challenges, including computational efficiency, data requirements, and generalization across domains. By analyzing the evolution and impact of Seq2Seq models, this review aims to highlight their significance in the field of deep learning and future research directions.[2]

2 FOUNDATIONS OF SEQUENCE-TO-SEQUENCE MODELS

2.1 HISTORICAL BACKGROUND AND EVOLUTION

To solve the sequence transformation problem where the length of the input and output sequences are different, Seq2Seq models were proposed. They provided effective approaches to learn mappings from sequences to sequences, thus were well-suited for tasks like machine translation, speech recognition and text generation. Often, the earliest Seq2Seq models were recurrent neural in nature, in which the encoder read the input sequence step-by-step and condensed it down into a fixed-length context vector (or hidden state). This vector was then fed to the decoder, which produced the output sequence token by token.[3]

However, this early approach suffered from several limitations, particularly the information bottleneck problem.

Encoding the entire input sequence into a single context vector led to poor performance when handling long sequences, as important information could be lost or diluted over time. This issue resulted in difficulties with capturing long-range dependencies, often causing degradation in translation quality or output coherence as sequence lengths increased.

To address these challenges, researchers introduced bidirectional encoders and attention mechanisms. Bidirectional RNNs (BiRNNs) allowed the encoder to process the input sequence in both forward and backward directions, providing richer context representations[1]. More importantly, the attention mechanism, first proposed by Bahdanau et al. (2015), revolutionized Seq2Seq models by enabling the decoder to dynamically focus on relevant parts of the input sequence at each decoding step.[4] Rather than relying solely on a fixed-length context vector, the attention mechanism computed a weighted sum of the encoder's hidden states, allowing the model to capture more nuanced dependencies and significantly improving translation accuracy and sequence generation quality.

The most significant breakthrough came with the introduction of the Transformer model by Vaswani et al. (2017). The Transformer architecture replaced recurrence with self-attention mechanisms, enabling models to process entire sequences in parallel rather than sequentially. This innovation led to substantial improvements in both efficiency and performance, particularly for long sequences[2]. Unlike RNN-based approaches, Transformers could capture long-range dependencies more effectively, scale better with large datasets, and benefit from extensive pretraining, leading to state-of-the-art performance in numerous sequence-based tasks.

The transition from traditional RNN-based Seq2Seq models to Transformer-based architectures has driven major advancements in natural language processing (NLP) and beyond. Transformer-based models such as BERT, GPT, T5, and BART have set new benchmarks in machine translation, text summarization, conversational AI, and other complex sequence tasks, demonstrating the lasting impact of Seq2Seq models and their evolution.

2.2 Core Architecture of Seq2Seq Models

The Seq2Seq Framework Seq2Seq is typically comprised of an encoder-decoder architecture: A high-level architecture that takes an input sequence and produces an output sequence with a different length. This architecture is commonly applied to sequence-to-sequence problems such as machine translation, text summarization, speech-to-text and conversational dialogue.

Encoder

The encoder is responsible for processing the input sequence and encoding it into a meaningful intermediate representation. Typically, it consists of a series of recurrent neural networks (RNNs), long short-term memory networks (LSTMs), gated recurrent units (GRUs), or Transformer-based layers. The encoder operates as follows:

Token Embedding: The input sequence is first converted into numerical representations through an embedding layer if working with textual data.

Contextual Representation: The sequence is then processed token by token, with each token's information being passed through multiple layers of the encoder.

Hidden States: The final hidden states capture the semantic and syntactic information of the input sequence.

Fixed-Length Context Vector (Traditional Approach): In early Seq2Seq models, the last hidden state of the encoder (a fixed-length context vector) was passed to the decoder. However, this caused performance degradation for long sequences due to loss of information.

Decoder

The decoder is responsible for generating the output sequence based on the encoded representation. It typically follows an autoregressive approach, where the model generates one token at a time while considering previously generated tokens. The decoder works as follows:

Receiving Initial Context: It starts with the context vector provided by the encoder.

Generating Output Step by Step: At each step, the decoder takes as input the previously generated token and updates its internal state.

Producing Predictions: The decoder applies a softmax layer to produce probabilities over possible next tokens.

Handling Variable Length Outputs: The generation process continues until a designated end-of-sequence (EOS) token is produced, indicating completion.

The encoder maps an input sequence $X = (x_1, x_2, \ldots, x_T)$ into a fixed-dimensional context vector h_T. It is typically implemented using recurrent neural networks (RNNs), long short-term memory (LSTM) networks, or gated recurrent units (GRUs). The encoder processes input tokens sequentially and updates a hidden state at each step:

$$h_t = f(W_h x_t + U_h h_{t-1} + b_h)$$

where h_t is the hidden state at time step t and W_h, U_h, b_h are learnable parameters.

The decoder generates the output sequence $Y = (y_1, y_2, \ldots, y_{T'})$ one token at a time. At each step, it takes the previous output and the context vector to produce the next token:

$$s_t = g(W_s y_{t-1} + U_s s_{t-1} + V_s h_T + b_s)$$

where s_t is the decoder's hidden state, and W_s, U_s, V_s, b_s are learnable parameters.

Academic Journal of Natural Science
Journal Home: http://ajns.suaspress.org | CODEN: AJNSAE
Vol. 2, No. 2, 2025 | ISSN 3078-5170 (Print) | ISSN 3078-5189 (Online) | ISSN 3078-5197 (Digital)

3 KEY TECHNIQUES AND ENHANCEMENTS IN SEQ2SEQ MODELS

3.1 Attention Mechanism

The fixed-length context vector in vanilla Seq2Seq models often struggles with long sequences. To mitigate this, attention mechanisms dynamically compute a weighted sum of encoder hidden states:

$$c_t = \sum_{i=1}^{T} \alpha_i^t h_i$$

where c_t is the attention-weighted context vector at decoding step t, and α_i^t represents attention weights, computed as:

$$\alpha_i^t = \frac{\exp(e_i^t)}{\sum_{j=1}^{T} \exp(e_j^t)}$$

where e_i^t is an alignment score that determines how much attention the decoder should pay to each encoder hidden state.

3.2 Transformers and Self-Attention

Traditional RNN-based Seq2Seq models suffer from sequential dependencies, making training inefficient. Transformers replace recurrence with self-attention, allowing each token to attend to all other tokens in parallel. The self-attention mechanism computes a weighted sum of input representations:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where Q, K, V are the query, key, and value matrices, and d_k is the key dimension. This approach significantly enhances computational efficiency and enables the model to capture long-range dependencies more effectively.

3.3 Advanced Architectures

Pitching Solutions to Seq2Seq Models: There have been a host of architectural improvements proposed to enhance Seq2Seq models; these address issues of information loss, repetitiveness, and incomplete content generation. Improvements to the model have made use of bidirectional encoders, pointer-generator networks, and copy and coverage mechanisms, leading to better performance across different sequence transformation tasks. [24]

One of the significant innovations in this framework was the bidirectional encoder, which processes the documentation series in ahead and backward directions and to seize richer contextual representation. With an inherent problem of traditional Seq2Seq models with unidirectional encoders, the models do not have sufficient context as it only takes into account the data until the sentence is being encoded. Whereas bidirectional recurrent neural networks (BiRNNs), bidirectional long short-term memory networks (BiLSTMs), and bidirectional gated recurrent units (BiGRUs) indeed provide the model to past and also to future, simultaneously. Bidirectional encoders Join (concatenate) the hidden states from both directions So, this enhances the understanding of Seq2Seq and makes more accurate and relevant outputs. [23]

Another significant enhancement is the pointer-generator network, which combines the strengths of both traditional Seq2Seq generation and direct copying of words from the input sequence. This is particularly useful in tasks like text summarization, dialogue generation, and question answering, where retaining specific words, such as names, numbers, or domain-specific terms, is crucial. Instead of relying solely on vocabulary-based generation, pointer-generator networks use a hybrid approach that allows the model to "point" to words in the input sequence and copy them into the output when necessary. This mechanism enhances factual accuracy and prevents the model from generating out-of-context or nonsensical outputs.[22]

Binded to the pointer mechanism, copy and coverage mechanisms are put forward to overcome two typical problems in Seq2Seq: repetition and missing content in text generation. By reusing the pointer-generator network which allows copying the words from the input for the previous words of the output, it ensures that some of the important information from the input is preserved. This is particularly advantageous in fields like legal text generation, summarization, and news writing, where certain terminology must be preserved. As a side note, the coverage mechanism helps to achieve no duplicate or overly repetitive words in generated text. It achieves this by maintaining a record of which parts of the input have been attended to already, making sure that all relevant information is included and minimizing the likelihood of over-generating specific words or phrases. This results in better quality summaries and text generation with clearer context.[21]

The integration of these techniques, along with the progression of attention structures and Transformer architectures, has led to remarkable improvements in Seq2Seq models. Apart from these techniques, modern Seq2Seq frameworks leverage bidirectional encoding, hybrid pointer-generator approaches and content-aware mechanisms to deliver high accuracy, coherence, and relevant outputs that have become crucial in any NLP based task.

3.4 Optimization Strategies

For the training of Seq2Seq models to be effective, we need to optimize the decoding phase so that out of it we can get high-quality outputs. Given that such models have problems such as exposure bias, potentially generating sub-optimal sequences, and employing weak search strategies,

there are numerous techniques introduced that improve both training and inference performance of such models. Some of the most commonly employed optimization strategies include beam search, scheduled sampling, and reinforcement learning techniques, which focus on different aspects of Seq2Seq model training and inference.

Beam Search: Beam search is a widely used decoding technique that enhances the generation of output sequences through beam width, where multiple possible outputs are considered at each decoding step. Beam search: While greedy decoding only outputs the most likely token at each time step, beam search stores beamwidth candidates sequences and grows them iteratively according to their cumulative probabilities. This allows beam search to perform better for longer sequences by sorting and discarding candidate sequences. Such models excel in fluency and contextuality which become essential for tasks such as machine translation, text summarization, and dialogue generation leading to higher outputs. But above larger beam widths, computational complexity explodes, and beam search may still lead to redundancy or bland outputs. In response to these concerns, various adaptations like length penalties, diversity-enforcing constraints, and diverse sampling techniques have been proposed to address diversity in results.

Another critical optimization technique is scheduled sampling, which helps mitigate exposure bias, a common issue in Seq2Seq models trained with teacher forcing. In standard training, the decoder is fed the correct previous token from the ground truth rather than its own predicted token, making it heavily reliant on perfect inputs. This dependency leads to poor performance during inference, where errors can accumulate and degrade the quality of generated sequences. Scheduled sampling addresses this by gradually transitioning the model from teacher-forced training to more autonomous generation. Initially, the model is trained using the correct ground-truth tokens, but as training progresses, a probability-based strategy determines whether the decoder should use the actual previous token or its own predicted token. By reducing reliance on teacher forcing over time, scheduled sampling helps the model learn to recover from its own mistakes, improving robustness during real-world inference. Variants of scheduled sampling, such as curriculum-based sampling and adaptive strategies, further enhance stability by dynamically adjusting the sampling probability based on model performance.

Beyond traditional optimization strategies, reinforcement learning (RL) techniques have also been explored to improve Seq2Seq model performance, particularly for tasks where evaluating outputs requires sequence-level metrics rather than token-level probabilities. Standard Seq2Seq training often relies on maximum likelihood estimation (MLE), which optimizes token-level cross-entropy loss but does not directly optimize for evaluation metrics like BLEU, ROUGE, or METEOR, commonly used in machine translation and summarization. Reinforcement learning addresses this limitation by treating sequence generation as a decision-making process, where the model is rewarded for generating high-quality outputs. Techniques such as REINFORCE, self-critical sequence training (SCST), and policy gradient methods optimize the model to maximize expected rewards based on predefined metrics. While RL-based optimization can significantly improve output quality, it is often challenging to train due to issues such as high variance in reward estimation and instability in gradient updates. To address these issues, hybrid approaches combining MLE pretraining with RL fine-tuning have been proposed, balancing stability and performance improvements.

With the incorporation of beam search, scheduled sampling, and reinforcement learning into Seq2Seq approach, the models has significantly Enhanced towards fluency, coherence and real world applicability. Hence the new architecture helps improve various metrics for significant sum-up and conversational AI systems.

4 TRAINING AND EVALUATION OF SEQ2SEQ MODELS

4.1 Data Preparation and Preprocessing

Effective training of Seq2Seq models requires careful preprocessing of data, as the quality of input data significantly impacts the model's ability to generate coherent and contextually relevant outputs. Since Seq2Seq models operate on structured input-output sequences, preprocessing plays a crucial role in handling out-of-vocabulary (OOV) words, managing long sequences, and ensuring consistency in training and inference. Among the key preprocessing techniques are tokenization strategies, sequence length management, and hierarchical encoding approaches, each designed to improve model generalization and efficiency.

One of the primary challenges in Seq2Seq training is dealing with out-of-vocabulary words, which can hinder model performance and lead to poor-quality outputs. Traditional word-level tokenization often results in a large vocabulary size, making it difficult for the model to handle rare or unseen words effectively. To address this, subword-based tokenization techniques such as byte pair encoding (BPE), wordpiece models, and unigram language models have been widely adopted. BPE, for example, breaks words into smaller frequently occurring subword units, allowing the model to dynamically compose rare words from known subword components. Similarly, wordpiece models, used in architectures like BERT and T5, segment words based on a statistical analysis of training data, ensuring a more balanced trade-off between vocabulary size and representation capacity. These techniques enhance generalization, reduce the need for large vocabulary embeddings, and improve robustness against OOV words, making them essential for Seq2Seq applications such as machine translation, text summarization, and speech-to-text systems.

Another critical preprocessing challenge in Seq2Seq training is handling long sequences, which can lead to issues such as increased computational costs, memory constraints, and difficulty in capturing long-range dependencies. Since Seq2Seq models often have a fixed input length due to hardware limitations, various strategies are employed to manage long sequences effectively. Truncation is commonly used to limit input length by discarding excess tokens beyond a predefined maximum length, ensuring efficient processing but at the risk of losing important information. Conversely, padding is applied to shorter sequences to maintain uniform input sizes, enabling batch training without disrupting the model's structural integrity. However, excessive padding can introduce redundant computations, so dynamic batching techniques are sometimes used to mitigate inefficiencies.

For tasks requiring long document processing, hierarchical encoding strategies have been developed to improve model efficiency and context retention. Instead of processing an entire long sequence at once, hierarchical encoders break the input into smaller segments, encode them separately, and then combine their representations to form a holistic understanding of the full sequence. This approach is particularly useful in document summarization, long-form question answering, and multi-turn dialogue generation, where context needs to be preserved across multiple sentences or paragraphs. Additionally, memory-augmented architectures and Transformer-based sparse attention mechanisms, such as Longformer and Reformer, provide alternative solutions for handling long sequences without incurring the computational overhead of standard self-attention mechanisms.

By leveraging effective tokenization techniques, sequence length management strategies, and hierarchical encoding approaches, Seq2Seq models can be trained more efficiently while improving their ability to generalize across different tasks. These preprocessing techniques play a crucial role in enhancing model robustness, reducing computational complexity, and ensuring high-quality sequence generation, making them indispensable in modern natural language processing applications.

4.2 Loss Functions and Training Objectives

The most common loss function for Seq2Seq training is cross-entropy loss, which measures the difference between predicted and ground-truth sequences:

$$L = -\sum_{t=1}^{T'} y_t \log \hat{y}_t$$

where $y_t$is the ground truth token at time step t, and $\hat{y}_t$ is the predicted probability of that token.

4.3 Evaluation Metrics

Evaluating Seq2Seq models is a crucial step in assessing their performance and ensuring that the generated sequences are both accurate and meaningful. Since these models are widely used in applications such as machine translation, text summarization, and dialogue generation, choosing the right evaluation metrics is essential for measuring output quality effectively. Various automated evaluation metrics exist, each with its strengths and limitations, and ongoing research continues to explore more sophisticated methods that better capture aspects such as fluency, coherence, and contextual relevance.

One of the most commonly used metrics for Seq2Seq evaluation is BLEU (Bilingual Evaluation Understudy), which measures n-gram overlap between the generated sequence and one or more reference sequences. BLEU assigns a score based on the proportion of overlapping unigrams, bigrams, trigrams, and higher-order n-grams, with a brevity penalty applied to prevent overly short outputs. While BLEU is widely used in machine translation, it has notable limitations—it primarily focuses on lexical similarity and does not account for semantic meaning, fluency, or syntactic correctness. As a result, a high BLEU score does not always guarantee that the generated text is well-formed or contextually appropriate. [25]

For tasks such as text summarization, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a more suitable metric. ROUGE measures recall-based text similarity by comparing the number of overlapping n-grams, word sequences, or even concepts between the generated and reference summaries. The most commonly used variants include ROUGE-N (n-gram overlap), ROUGE-L (longest common subsequence), and ROUGE-S (skip-bigram overlap). ROUGE is particularly useful in evaluating summarization models because it assesses how much of the reference content is retained in the generated summary. However, similar to BLEU, it does not fully capture fluency, coherence, or informativeness, leading to discrepancies between automated scores and human judgments.

Since both BLEU and ROUGE are based on surface-level lexical matching, they often fail to assess the true semantic meaning of a generated sequence. To address this limitation, newer evaluation metrics such as METEOR, BERTScore, and MoverScore have been introduced. METEOR (Metric for Evaluation of Translation with Explicit ORdering) improves upon BLEU by incorporating synonym matching, stemming, and paraphrase recognition, making it more robust for evaluating machine translation and text generation tasks. BERTScore, on the other hand, leverages pre-trained transformer models like BERT to compare embeddings of generated and reference sentences, allowing for a more context-aware evaluation that captures semantic similarity beyond word overlap. MoverScore, an extension of BERTScore, computes the optimal transport distance between embeddings of words in the generated and reference

texts, further enhancing evaluation quality by considering contextual relationships.

Despite advancements in automated metrics, they still fall short in fully capturing fluency, coherence, factual consistency, and contextual appropriateness. This has led to ongoing research in human evaluation methods, where expert annotators or crowd-sourced workers assess text quality based on criteria such as grammatical correctness, readability, logical consistency, and informativeness. Human evaluation is often considered the gold standard, but it is time-consuming, expensive, and prone to subjectivity. As an alternative, researchers are developing learned evaluation metrics, where neural models are trained on human-labeled evaluation datasets to predict human-like scores for generated text. Examples include BLEURT (a learned evaluation metric based on BERT) and COMET (a deep learning-based metric for machine translation evaluation), which outperform traditional metrics by aligning more closely with human judgments.

5 APPLICATIONS AND REAL-WORLD USE CASES

5.1 NATURAL LANGUAGE PROCESSING APPLICATIONS

Seq2Seq models have revolutionized various natural language processing (NLP) tasks, enabling significant advancements in machine translation, text summarization, and conversational AI. By leveraging their encoder-decoder architecture and integrating improvements such as attention mechanisms and Transformer-based models, Seq2Seq systems have become the backbone of many real-world applications, delivering more fluent, accurate, and context-aware outputs. Their versatility has made them indispensable in domains ranging from automated content generation to speech processing and dialogue systems.

One of the most impactful applications of Seq2Seq models is machine translation (MT). Early neural machine translation (NMT) systems, such as those used in Google Translate, DeepL, and Microsoft Translator, initially relied on RNN-based Seq2Seq models with attention mechanisms to improve translation accuracy. However, the introduction of the Transformer model significantly enhanced performance by allowing models to capture long-range dependencies more effectively. Today, state-of-the-art NMT systems use pre-trained language models such as mBART, MarianMT, and M2M-100, which are trained on large multilingual datasets and fine-tuned for specific translation tasks. These models enable context-aware, high-quality translations across hundreds of languages, making machine translation more robust and scalable than ever before.

Another critical application of Seq2Seq models is text summarization, which is broadly categorized into extractive and abstractive summarization. Extractive summarization selects key sentences from a document and rearranges them to form a concise summary, while abstractive summarization generates a new, more natural summary by paraphrasing the original content. Seq2Seq models have been particularly successful in abstractive summarization, where they generate summaries that maintain the key ideas while improving readability. Systems like Google's Pegasus, OpenAI's GPT-based summarizers, and BART are designed to generate high-quality summaries for news articles, research papers, and legal documents. These models use pre-training techniques such as gap-sentence generation (GSG) and masked language modeling (MLM) to learn how to generate coherent and concise summaries. Industries such as journalism, business intelligence, and academic research benefit greatly from these advances, as automated summarization reduces information overload and helps users quickly grasp key insights.

Seq2Seq models are also at the core of chatbots and conversational AI, powering virtual assistants such as Google Assistant, Amazon Alexa, and OpenAI's ChatGPT. Unlike rule-based chatbots, which rely on predefined scripts, Seq2Seq-based conversational models can generate dynamic, contextually relevant responses based on the input dialogue history. Early conversational AI models faced challenges such as generic or incoherent responses, but modern systems integrate contextual embeddings, dialogue history retention, and reinforcement learning to improve user interactions. Transformer-based models such as DialoGPT, Meena, and LaMDA leverage large-scale conversational data to generate more engaging and human-like responses, enhancing applications in customer service, virtual tutoring, and interactive storytelling. These advancements have led to the development of emotion-aware and multi-turn dialogue systems, making AI-driven conversations more personalized and context-sensitive [3].

In addition to their application in text-based systems, Seq2Seq networks are also widely deployed in various speech processing tasks; speech-to-text (ASR, automatic speech recognition) and text-to-speech (TTS) synthesis. ASR systems (example Google Speech-to-Text, Whisper by OpenAI, DeepSpeech) use Seq2Seq architectures for transforming spoken language into text, which have enabled real-time transcription service, voice assistants and automated captions. TTS systems like Tacotron 2 and FastSpeech convert written text into natural-sounding speech, enhancing applications in audiobooks, assistive technology, and virtual avatar voice synthesis [4].

Seq2Seq models have also been applied in code generation and programming assistants, where they help in tasks such as code completion, bug fixing, and natural language to code translation [5]. Models like Codex (the foundation of GitHub Copilot) and AlphaCode utilize Seq2Seq architectures to understand and generate programming code, assisting developers in writing efficient and syntactically correct code snippets.

5.2 Speech and Audio Processing

Seq2Seq models have significantly advanced the field of speech and audio processing, enabling applications such as automatic speech recognition (ASR), text-to-speech (TTS) synthesis, voice translation, and speech-based conversational AI [6]. By leveraging their encoder-decoder architecture, these models can efficiently process variable-length audio sequences and generate accurate textual or spoken outputs [7]. The integration of attention mechanisms, Transformer-based architectures, and self-supervised learning has further improved their performance, making speech-based AI systems more robust, accurate, and natural-sounding.

One of the most important applications of Seq2Seq models in speech processing is automatic speech recognition (ASR), where audio waveforms are converted into text. Traditional ASR systems relied on separate components, including acoustic models, language models, and phonetic lexicons, but modern Seq2Seq-based approaches unify these processes into a single end-to-end trainable system. Models such as DeepSpeech, Wav2Vec 2.0, Whisper by OpenAI, and Conformer-based architectures leverage Seq2Seq techniques to improve ASR performance by directly mapping speech features to text outputs [8]. These models have been widely adopted in automated transcription services, real-time captioning, virtual assistants, and call center automation. For example, Google and Microsoft integrate Seq2Seq-based ASR models into their products to enable accurate speech recognition across multiple languages and accents.[21]

In addition to ASR, Seq2Seq models have transformed text-to-speech (TTS) synthesis, where textual input is converted into natural-sounding speech [9]. Early TTS systems relied on concatenative or parametric speech synthesis, which often resulted in robotic or unnatural outputs. Seq2Seq-based TTS models, such as Tacotron 2, FastSpeech, and VITS (Variational Inference Text-to-Speech), have significantly improved speech synthesis by generating high-quality, human-like speech with natural prosody and intonation [10]. These models use attention mechanisms to align text with speech features, producing fluid and expressive speech. TTS technology is widely used in virtual assistants (e.g., Alexa, Google Assistant, Siri), audiobook generation, assistive speech devices, and voice cloning applications [11].

Another exciting application of Seq2Seq models in speech processing is speech-to-speech translation (S2ST), where spoken language is directly translated into another spoken language without requiring intermediate text transcription[12]. Traditional machine translation pipelines involved speech recognition, text translation, and speech synthesis, but modern Seq2Seq-based S2ST models, such as Meta's SeamlessM4T and Google's Translatotron, streamline this process into a single model. These advancements have profound implications for real-time multilingual communication, global accessibility, and cross-lingual conversational AI[13].

Seq2Seq models have also contributed to speech-based conversational AI, where models are trained to understand and generate spoken responses [14]. Virtual assistants like Google Assistant, Amazon Alexa, and Apple's Siri rely on Seq2Seq-based speech processing to interpret spoken queries, generate relevant responses, and maintain contextual awareness in conversations. Advanced architectures, such as wav2vec 2.0 for speech encoding and GPT-based models for response generation, allow conversational AI systems to handle complex dialogue interactions, recognize emotions, and provide more natural, human-like interactions. [15]

Beyond mainstream applications, Seq2Seq models are also being used in voice biometrics, speaker diarization, emotion recognition, and speech enhancement. [16] For instance, speaker recognition systems use Seq2Seq-based models to identify and authenticate users based on their voiceprints, while speech enhancement models improve audio clarity in noisy environments, benefiting applications such as hearing aids, voice communication, and noise-canceling systems.[17]

6 CONCLUSION

Sequence-to-sequence models have significantly advanced the field of sequence modeling, transforming how machines process and generate complex sequences across diverse domains. [18] From early RNN-based implementations to the powerful Transformer architectures of today, Seq2Seq models have evolved to address limitations such as long-range dependency modeling and information bottlenecks. Enhanced by techniques like attention mechanisms, pointer-generator networks, and advanced optimization strategies, these models have become essential in applications such as machine translation, summarization, and speech recognition. [19] Despite their impressive achievements, challenges remain, including improving generalization, reducing computational cost, and ensuring output quality. Continued research into hybrid architectures, efficient training methods, and semantically-aware evaluation metrics will further extend the capabilities and impact of Seq2Seq systems in the future of AI. [20]

ACKNOWLEDGMENTS

The authors thank the editor and anonymous reviewers for their helpful comments and valuable suggestions.

FUNDING

Not applicable.

INSTITUTIONAL REVIEW BOARD STATEMENT

Not applicable.

INFORMED CONSENT STATEMENT

Not applicable.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

CONFLICT OF INTEREST

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

PUBLISHER'S NOTE

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

AUTHOR CONTRIBUTIONS

Not applicable.

ABOUT THE AUTHORS

BO, Tao

TikTok.Inc, USA.

LI, Weiyi

Georgia Institute of Technology, USA.

LIU, Yue

TikTok.Inc, USA.

REFERENCES

[1] Lin, W., Xiao, J., & Cen, Z. (2024). Exploring Bias in NLP Models: Analyzing the Impact of Training Data on Fairness and Equity. Journal of Industrial Engineering and Applied Science, 2(5), 24-28.

[2] Katrompas, A., Ntakouris, T., & Metsis, V. (2022, June). Recurrence and self-attention vs the transformer for time-series classification: A comparative study. In International Conference on Artificial Intelligence in Medicine (pp. 99-109). Cham: Springer International Publishing.

[3] Lin, C. C., Huang, A. Y., & Yang, S. J. (2023). A review of ai-driven conversational chatbots implementation methodologies and challenges (1999–2022). Sustainability, 15(5), 4012.

[4] Lin, W. (2024). The Application of Real-time Emotion Recognition in Video Conferencing. Journal of Computer Technology and Applied Mathematics, 1(4), 79-88.

[5] Dong, Y., Li, G., & Jin, Z. (2023, July). CODEP: grammatical seq2seq model for general-purpose code generation. In Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis (pp. 188-198).

[6] Li, K., Chen, X., Song, T., Zhou, C., Liu, Z., Zhang, Z., Guo, J., & Shan, Q. (2025a, March 24). Solving situation puzzles with large language model and external reformulation.

[7] Lyu, S. (2024). Machine Vision-Based Automatic Detection for Electromechanical Equipment. Journal of Computer Technology and Applied Mathematics, 1(4), 12-20.

[8] Barakat, H., Turk, O., & Demiroglu, C. (2024). Deep learning-based expressive speech synthesis: a systematic review of approaches, challenges, and resources. EURASIP Journal on Audio, Speech, and Music Processing, 2024(1), 11.

[9] Lin, W. (2025). Enhancing Video Conferencing Experience through Speech Activity Detection and Lip Synchronization with Deep Learning Models. Journal of Computer Technology and Applied Mathematics, 2(2), 16-23.

[10] Orynbay, L., Razakhova, B., Peer, P., Meden, B., & Emeršič, Ž. (2024). Recent advances in synthesis and interaction of speech, text, and vision. Electronics, 13(9), 1726.

[11] Luo, M., Zhang, W., Song, T., Li, K., Zhu, H., Du, B., & Wen, H. (2021, January). Rebalancing expanding EV sharing systems with deep reinforcement learning. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence (pp. 1338-1344).

[12] Lyu, S. (2024). The Application of Generative AI in Virtual Reality and Augmented Reality. Journal of Industrial Engineering and Applied Science, 2(6), 1-9.

[13] Lin, W. (2024). A Systematic Review of Computer Vision-Based Virtual Conference Assistants and Gesture Recognition. Journal of Computer Technology and Applied Mathematics, 1(4), 28-35.

[14] Zhu, H., Luo, Y., Liu, Q., Fan, H., Song, T., Yu, C. W., & Du, B. (2019). Multistep flow prediction on car-sharing systems: A multi-graph convolutional neural network with attention mechanism. International Journal of Software Engineering and Knowledge Engineering, 29(11n12), 1727–1740.

[15] Lyu, S. (2024). The Technology of Face Synthesis and Editing Based on Generative Models. Journal of Computer Technology and Applied Mathematics, 1(4), 21-27.

[16] Xu, F. (2021). Gaozhi yuanxiao yingyu jiaoxue moshi chuangxin [Innovation of English teaching models in higher vocational colleges]. Jiuzhou Press.

[17] Li, K., Chen, X., Song, T., Zhang, H., Zhang, W., & Shan, Q. (2024). GPTDrawer: Enhancing Visual Synthesis through ChatGPT. arXiv preprint arXiv:2412.10429.

[18] Jia, Y., Weiss, R. J., Biadsy, F., Macherey, W., Johnson, M., Chen, Z., & Wu, Y. (2019). Direct speech-to-speech translation with a sequence-to-sequence model. arXiv preprint arXiv:1904.06037.

[19] Lin, W. (2024). A Review of Multimodal Interaction Technologies in Virtual Meetings. Journal of Computer Technology and Applied Mathematics, 1(4), 60-68.

[20] Li, X., Wang, X., Qi, Z., Cao, H., Zhang, Z., & Xiang, A. DTSGAN: Learning Dynamic Textures via Spatiotemporal Generative Adversarial Network. Academic Journal of Computing & Information Science, 7(10), 31-40.

[21] Gholami, M. J., & Al Abdwani, T. (2024). The rise of thinking machines: A review of artificial intelligence in contemporary communication. Journal of Business, Communication & Technology, 1-15.

[22] Luo, M., Du, B., Zhang, W., Song, T., Li, K., Zhu, H., ... & Wen, H. (2023). Fleet rebalancing for expanding shared e-Mobility systems: A multi-agent deep reinforcement learning approach. IEEE Transactions on Intelligent Transportation Systems, 24(4), 3868-3881.

[23] Li, X., Cao, H., Zhang, Z., Hu, J., Jin, Y., & Zhao, Z. (2024). Artistic Neural Style Transfer Algorithms with Activation Smoothing. arXiv preprint arXiv:2411.08014.

[24] Sun, Y., & Ortiz, J. (2024). An ai-based system utilizing iot-enabled ambient sensors and llms for complex activity tracking. arXiv preprint arXiv:2407.02606.

[25] Gligorea, I., Cioca, M., Oancea, R., Gorski, A. T., Gorski, H., & Tudorache, P. (2023). Adaptive learning using artificial intelligence in e-learning: A literature review. Education Sciences, 13(12), 1216.