A Survey on Neural Approaches to Natural Language Generation in Open-Domain Dialogue Systems

Adam Nik

CGSI130: What are Minds and What They do

Carleton College

A Survey on Neural Approaches to Natural Language Generation in Open-Domain Dialogue Systems

Introduction

Alan Turing, one of the earliest pioneers in the field of artificial intelligence (AI), begged the question of whether a machine could exhibit human-levels of intelligence (Turing, 1950). Turing believed that if a computer could imitate human-levels of language and could produce language to the degree where its generated utterances were indistinguishable from true human dialogue, then it has exhibited human-levels of thinking ability. Turing's work has since sparked decades of research in the fields of AI and natural language processing (NLP) to develop dialogue systems to pass his Turing Test. This survey paper reviews the modern day advances of dialogue systems, with a focus on the natural language generation of open-domain dialogue systems.

Natural language generation is the task of creating text from underlying, non-linguistic representation of information (C. Dong et al., 2021), and is considered a sub-field of NLP, AI, and cognitive science (Santhanam & Shaikh, 2019). Natural language generation is often broken down into three categories: text abbreviation, text expansion, and text rewriting and reasoning (C. Dong et al., 2021). We will focus on the task of text reasoning in this paper, which is defined as applying reasoning methods to textual information to generate responses, as it is the main type of text generation used in dialogue systems.

Dialogue systems, as known as conversational agents, are computer programs that are designed to interact with human users through natural speech or text so that the user thinks they are having dialogue with a real human (Hussain, Ameri Sianaki, & Ababneh, 2019). The earliest versions of dialogue systems, such as ELIZA (Weizenbaum, 1966) and PARRY (Colby, 1975), were rule-based systems that used pattern matching to produce responses (Adamopoulou & Moussiades, 2020). However, those approaches were rigid and unable to produce original utterances. As a result, modern day approaches have shifted

towards retrieval-based, generative, or ensemble systems. Retrieval-based approaches are similar to rule-based approaches but differentiate from them as the system retrieves response candidates from a separate database before applying the matching approach to the response selection(Wu, Wu, Xing, Zhou, & Li, 2017). Generative approaches use language models or seq2seq architectures to produce free-form responses (Santhanam & Shaikh, 2019). Finally, ensemble systems combine generative methods and retrieval-based methods by either comparing retrieved and generated responses or using generative models to refine the retrieved responses (I. V. Serban et al., 2017; Zhu, Cui, Zhang, Wei, & Liu, 2019). Furthermore, dialogue systems can be categorized into task-oriented and open-domain systems. Task-oriented dialogue systems solve specific problems in a certain domain such as movie ticket booking, restaurant table reserving, etc (Ni, Young, Pandelea, Xue, & Cambria, 2022). Open-domain systems, or chat-bots, are conversational agents that are not restricted to a certain task or domain, which allows a higher focus towards generating dialogues with high similarity to how humans converse (J. Li et al., 2017).

Open-domain systems better represent the envisioned language systems studied by Turing (1950) and are thus the primary focus of this review. The following sections of this paper will outline state-of-the-art neural approaches to language generation and then comment on the current challenges surrounding language generation in open-domain dialogue systems.

Neural Approaches to Natural Language Generation

Neural Networks are a class of machine learning models that are capable of identifying patterns in text and identify features that help solve a variety of classification and generative problems (Sutskever, Vinyals, & Le, 2014) and have become the state-of-the-art in artificial intelligence. They are complex, mathematical models comprised of node layers that are designed to learn representations at increasing levels of abstraction by exploiting back-propagation (Gatt & Krahmer, 2018; LeCun, Bengio, & Hinton, 2015).

The paradigm of using multiple layers of processing to extract progressively higher level features from data is referred to as *Deep Learning* (LeCun et al., 2015).

Feed-forward Neural Networks

The simplest and most widely used type of neural network is the feed forward neural network (FFN) or multi-layer perceptron (Rosenblatt, 1958). FFNs first gained popularity in the 1960s and inspired many of the connectionist architectures of cognitive science, such as Rumelhart (1989).

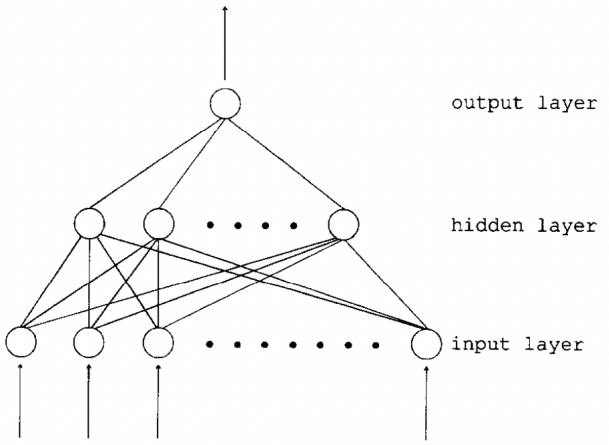

Figure 1: A simple feed forward network architecture from Svozil, Kvasnicka, and Pospichal (1997).

A diagram of a FFN is shown above in Figure 1. Each layer of the FFN can be defined as

$$y = \sigma(Wx + b) \tag{1}$$

where σ is a non-linear activation function and W and b are trainable parameters (Ni et al., 2022). Despite being the building blocks of all state-of-the-art neural approaches, simple FFNs fall short to modern approaches as they 1) assume data points are independent and therefore miss important information when dealing with interrelated data points and 2) can typically only handle inputs with fixed length, which is a limitation when processing sequential data varying in length (Lipton, Berkowitz, & Elkan, 2015).

Recurrent Neural Networks (RNNs)

First proposed by Elman (1990), recurrent neural networks (RNNs) are models that process sequential data with the help of recurrent connections that perform the same task over every sequence (I. Goodfellow, Bengio, & Courville, 2016). RNNs get their name from their *recurrent connections*, which are feedback loops that cycle throughout the model (as shown in Figure 2). Due to their architecture, RNNs have the concept of "memory" that allows them to store information of previous inputs or states to use when generating the next output of the sequence.

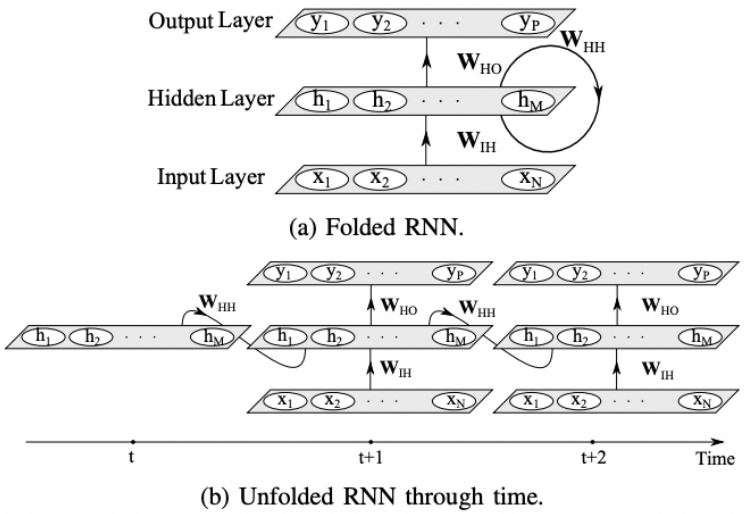

Figure 2: An RNN architecture from Salehinejad, Sankar, Barfett, Colak, and Valaee (2017), showing both the A) Folded and 2) Unfolded versions of the architecture.

In practice, RNNs suffer from gradient explosion and vanishing in their back-propagation, thus making accurate outputs difficult to achieve. To combat this, methods such as long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent units (GRU) (Chung, Gulcehre, Cho, & Bengio, 2014) are used to improve the output calculations of the RNN's hidden layers.

Seq2Seq Architecture

First proposed by Sutskever et al. (2014), Sequence to Sequence, or seq2seq, architectures are designed to solve modelling problems where both the input and intended output are sequences. A seq2seq model typically consists of two RNNs (in recent years, researchers have shifted more towards using Transformers) in an encoder-decoder structure with the goal of mapping input text to output text. Due to the format of the architecture, seq2seq models are also popular in applications such as machine translation (Chen et al., 2018) and image to text language generation (Zhou et al., 2020) The encoder of the model maps an input sequence to a vector of a fixed dimensionality, and the decoder produces the target output sequence from that encoded vector.

Transformers

Transformers are a relatively new neural architecture and have become the state-of-the-art in many tasks within the field of NLP. Proposed by Vaswani et al. (2017), transformers are encoder-decoder models that do away with recurrent connections and instead leverage attention mechanisms to learn the context of the sequential input. The attention mechanisms allow the the model to choose which parts of the input are most important to producing an output and thus gain higher emphasis. The self-attention of the Transformer derives the relationships between each given token in the input sequence and the other tokens of the input, and is computed as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}}) \tag{2}$$

where Q, K, and V are all matrices computed from the input embeddings, and d_k is used as a scaling factor. Additionally, the Transformer makes use of multi-head attention, which works to extract multiple features from the input.

$$\begin{aligned} &MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^o, \\ &head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \tag{3}$$

where each W matrix is a trainable matrix. The full model architecture is provided below:

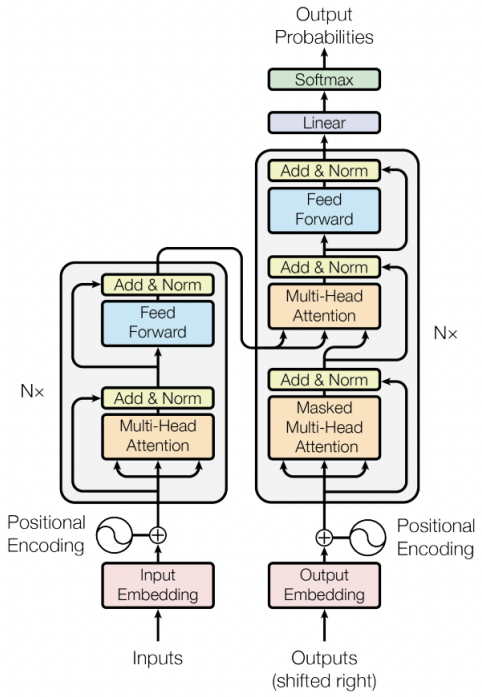

Figure 3: The Transformer Architecture, from Vaswani et al. (2017).

One of the most prominent improvements of Transformers over RNNs is the Transformer's parallelism. Unlike the calculations of the RNN, where each $t + 1$ token computation depends on the hidden-layer output of the previous t token, the Transformer is able to compute its attention mechanisms independently of one another, and thus is viable for parallelization. Thus, with the help of modern day GPUs, Transformer models are able to be trained much faster with much larger parameter spaces and much more training data.

Pre-training

Due to the less costly training process of Transformers, research in recent years has focused on training large models on large corpora of unlabeled data with the target task of self-supervised task such as cloze task prediction (Taylor, 1953) and next sentence prediction (Devlin, Chang, Lee, & Toutanova, 2018). The act of training a large language

model on these general tasks is called *pre-training*, which is the pre-requisite to *fine-tuning*, which is re-training the final layers to the model on labeled data for a specific task. Radford, Narasimhan, Salimans, Sutskever, et al. (2018) show that pre-training a model on unlabeled text vastly improves task performance on the down-stream fine-grain tasks, as the model is able to learn general language features from the large unlabeled text and use that feature detection in the specific tasks.

Many state-of-the-art pre-trained models such as BERT (Devlin et al., 2018), the GPT series (Brown et al., 2020; Radford et al., 2018, 2019), and UniLM (L. Dong et al., 2019) have been released within the past few years and have made large-scale language models more accessible than ever. Several current approaches to dialogue systems make use of these pre-trained models in their language generation pipelines. For example, Z. Li et al. (2019) created a system that uses multiple incremental transformer encoders to encode multi-turn conversations and their related document knowledge. Similarly, Bao, He, Wang, Wu, and Wang (2020) propose PLATO, a dialogue generation pre-training framework that utilizes a stacked transformer structure.

Reinforcement Learning Approaches

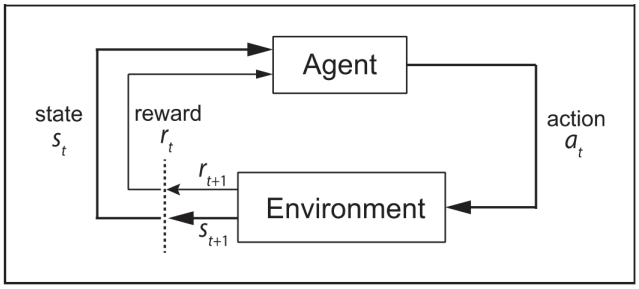

Figure 4: The Reinforcement Learning Paradigm. Diagram from Galatzer-Levy, Ruggles, and Chen (2018).

Reinforcement learning (RL) is a paradigm of machine learning that aims to teach a given agent how to best perform in its environment based on the rewards it receives from

its actions. RL problems are formulated as Markov Decision Processes (MDPs) tuples $\langle S, A, T, R, \gamma \rangle$ where S is the set of environment states, A is the set of possible actions available to the agent, T is the transition probability function of one state to another given an action $T : S \times A \times S \rightarrow [0, 1]$, R is the reward received from moving from one state to another $R : S \times A \times S \rightarrow R$, and $\gamma \in [0, 1)$ is a discount factor, which quantifies how much importance we give for future rewards (Sutton & Barto, 2018). Given this setup, the goal of the training process is to find a policy $\pi(a|s)$ that maximizes the expected rewards received from the environment (Luketina et al., 2019). The policy can be seen as a guide for the agent to know what actions to take in a given environment state.

The RL framework has been argued to be better at handling uncertainty in dynamic environments than supervised learning or classification, as RL allows for adaptation in a changing context (Rieser & Lemon, 2009). As a result, RL approaches are well suited to solve many challenges in dialogue systems as we can frame the dialogue system as the agent and the task of language generation in conversation as the environment. Many of the state-of-the-art RL-based dialogue systems are ensemble-based systems that use RL algorithms to select the best candidate response from pooled retrieval-based and generative-based responses (I. V. Serban et al., 2017; Zhu et al., 2019).

Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) are a generative modelling framework that learns to produce output through an adversarial training process between two models, a generator and a discriminator. First introduced by (I. J. Goodfellow et al., 2014), GANs are implemented using deep learning models such as convolutional neural networks (LeCun et al., 1989; LeCun, Bottou, Bengio, & Haffner, 1998) or Transformers. Within the framework, the generator and discriminator models are simultaneously trained and the training process can be viewed as a competition between them. The generator is trained with the task of generating new samples that are alike to the original training data while

the discriminator attempts to distinguish between original data (real) and generated data (fake). Within the framework of dialogue agents, GANs are used to enhance response generation as the discriminator distinguishes generated responses from human responses, which incentivizes the agent, which is also the generator in GAN, to generate higher-quality responses (Ni et al., 2022).

Challenges of Language Generation in Open-Domain Dialogue Systems

Being able to produce human-level quality responses in dialogue systems is a difficult task due to the intrinsic components apparent in human conversation, and the massive output space needed to consider the vast amount of possible responses. This is especially relevant to open-domain systems as opposed to task-oriented systems as task-oriented systems are far more rigid in their dialogue and are confined to their pre-defined tasks. In this section, we will touch on three of the main challenges surrounding natural language generation in open-domain systems, which include response coherence, context awareness, and diversity in responses.

Response Coherence

Response coherence refers to maintaining logic and consistency throughout the course of a conversation. Thus, dialogue agents often need some form of internal evaluation mechanism that regulates the quality of the system's responses. Recently, Bao, He, Wang, Lian, and Wu (2019) proposed a Generation-Evaluation framework that evaluated the several qualities of the system's responses, including coherence. The feedback was used as a reward signal in the RL framework that guided the system to a better dialogue strategy, thus improving the response quality. Additionally, Zhu et al. (2019) built a retrieval-enhanced generation model based on a GAN architecture. First, a discriminator was trained with the help of a retrieval system, and the generator was trained under the supervision signal of a discriminator. Second, the retrieved responses were also used as a part of the generator input to provide a coherent example for the generator.

Context Awareness

Context awareness means being able to remember information from previous parts of the dialogue; context awareness is important because a system's response should be based on all information present in a dialogue history, not just the last available message. Lack of context awareness was an obstacle to earlier dialogue systems that approached dialogue generation as a one-to-one task and only used the last utterance as input. Sordoni et al. (2015) approach the issue of context awareness by summing representations of previous turns with the current turn utterance. More recently, I. Serban, Sordoni, Bengio, Courville, and Pineau (2016) use a Hierarchical Recurrent Encoder–Decoder (HRED) architecture to build an end-to-end context-aware dialogue system; the HRED learned both token-level and turn-level representation in order to compute conversational context.

Diversity of Responses

Due to dialogue systems being trained on data distributions of human language corpora, they tend to favor common, yet dull responses such as "*okay*" or "*I don't know*". As a result, a strong emphasis has been placed on prioritizing more unique response utterances that better emulate how a human would perform in conversation. Some works further use a re-ranking stage in their language generation module to select more diverse responses in the generated N-best list. For example, Qiu, Li, Bi, Zhao, and Yan (2019) proposed a two-stage generation model to increase response where the first stage extracts common features of multiple ground truth responses and the second stage selects the most distinctive one as the final response. Other recent approaches toward conversational modeling have tackled the issue of dull and generic response through the use of previous contextual information, attention mechanism, or reinforcement learning that penalizes the agent when it produces trivial or repetitive utterance (J. Li et al., 2016, 2017; Liu et al., 2018).

Conclusion

In this paper, we reviewed the state-of-the-art of natural language generation in open-domain dialogue systems, with a focus on neural approaches to generation. Motivated by Turing (1950)'s speculations that machines can think, we set out to see how far machines have come towards the goal of truly imitating human language. We discussed simple feed-forward networks, RNNs, Sequence to Sequence architectures, Transformers, pre-training in language modeling, Reinforcement Learning in language generation, and finally Generative Adversarial Networks. Additionally, we assessed the most prevalent challenges in language generation when it comes to open-domain systems and emulating true, human-like dialogue. As both general AI and dialogue systems have continued to progress dramatically in recent decades, we believe that it is almost inevitable that dialogue systems will eventually reach a state of being indistinguishable from real humans.

References

Adamopoulou, E., & Moussiades, L. (2020). An overview of chatbot technology. In *Ifip international conference on artificial intelligence applications and innovations* (pp. 373–383).

Bao, S., He, H., Wang, F., Lian, R., & Wu, H. (2019). Know more about each other: Evolving dialogue strategy via compound assessment. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 5382–5391).

Bao, S., He, H., Wang, F., Wu, H., & Wang, H. (2020). Plato: Pre-trained dialogue generation model with discrete latent variable. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 85–96).

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... others (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877–1901.

Chen, M. X., Firat, O., Bapna, A., Johnson, M., Macherey, W., Foster, G., ... others (2018). The best of both worlds: Combining recent advances in neural machine translation. In *Acl (1)*.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *Nips 2014 workshop on deep learning, december 2014*.

Colby, K. M. (1975). *Artificial paranoia: a computer simulation of paranoid process*. Pergamon Press.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, C., Li, Y., Gong, H., Chen, M., Li, J., Shen, Y., & Yang, M. (2021). A survey of natural language generation. *arXiv e-prints*, arXiv–2112.

Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., ... Hon, H.-W. (2019). Unified

language model pre-training for natural language understanding and generation. *Advances in Neural Information Processing Systems*, *32*.

Elman, J. L. (1990). Finding structure in time. *Cognitive science*, *14*(2), 179–211.

Galatzer-Levy, I. R., Ruggles, K. V., & Chen, Z. (2018). Data science in the research domain criteria era: relevance of machine learning to the study of stress pathology, recovery, and resilience. *Chronic Stress*, *2*, 2470547017747553.

Gatt, A., & Krahmer, E. (2018). Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, *61*, 65–170.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. (`http://www.deeplearningbook.org`)

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. In *Proceedings of the 27th international conference on neural information processing systems-volume 2* (pp. 2672–2680).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hussain, S., Ameri Sianaki, O., & Ababneh, N. (2019). A survey on conversational agents/chatbots classification and design techniques. In *Workshops of the international conference on advanced information networking and applications* (pp. 946–956).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., & Jackel, L. (1989). Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, *2*.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324.

Li, J., Monroe, W., Ritter, A., Jurafsky, D., Galley, M., & Gao, J. (2016). Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 1192–1202).

Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A., & Jurafsky, D. (2017). Adversarial learning for neural dialogue generation. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2157–2169).

Li, Z., Niu, C., Meng, F., Feng, Y., Li, Q., & Zhou, J. (2019). Incremental transformer with deliberation decoder for document grounded conversations. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 12–21).

Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*.

Liu, Y., Bi, W., Gao, J., Liu, X., Yao, J., & Shi, S. (2018). Towards less generic responses in neural conversation models: A statistical re-weighting method. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 2769–2774).

Luketina, J., Nardelli, N., Farquhar, G., Foerster, J., Andreas, J., Grefenstette, E., ... Rocktäschel, T. (2019). A survey of reinforcement learning informed by natural language. *arXiv preprint arXiv:1906.03926*.

Ni, J., Young, T., Pandelea, V., Xue, F., & Cambria, E. (2022). Recent advances in deep learning based dialogue systems: A systematic survey. *Artificial Intelligence Review*, 1–101.

Qiu, L., Li, J., Bi, W., Zhao, D., & Yan, R. (2019). Are training samples correlated? learning to generate dialogue responses with multiple references. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 3826–3835).

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019).

Language models are unsupervised multitask learners.

Rieser, V., & Lemon, O. (2009). Natural language generation as planning under uncertainty for spoken dialogue systems. In *Empirical methods in natural language generation* (pp. 105–120). Springer.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, *65*(6), 386.

Rumelhart, D. E. (1989). The architecture of mind: A connectionist approach.

Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2017). Recent advances in recurrent neural networks. *arXiv preprint arXiv:1801.01078*.

Santhanam, S., & Shaikh, S. (2019). A survey of natural language generation techniques with a focus on dialogue systems-past, present and future directions. *arXiv preprint arXiv:1906.00500*.

Serban, I., Sordoni, A., Bengio, Y., Courville, A., & Pineau, J. (2016). Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 30).

Serban, I. V., Sankar, C., Germain, M., Zhang, S., Lin, Z., Subramanian, S., ... others (2017). A deep reinforcement learning chatbot. *arXiv preprint arXiv:1709.02349*.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., ... Dolan, W. B. (2015). A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 196–205).

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, *27*.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Svozil, D., Kvasnicka, V., & Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Chemometrics and intelligent laboratory systems*, *39*(1), 43–62.

Taylor, W. L. (1953). "cloze procedure": A new tool for measuring readability. *Journalism quarterly*, *30*(4), 415–433.

Turing, A. M. (1950). Mind. *Mind*, *59*(236), 433–460.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Weizenbaum, J. (1966). Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, *9*(1), 36–45.

Wu, Y., Wu, W., Xing, C., Zhou, M., & Li, Z. (2017). Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 496–505).

Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J., & Gao, J. (2020). Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 34, pp. 13041–13049).

Zhu, Q., Cui, L., Zhang, W.-N., Wei, F., & Liu, T. (2019, July). Retrieval-enhanced adversarial training for neural response generation. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 3763–3773). Florence, Italy: Association for Computational Linguistics. Retrieved from https://aclanthology.org/P19-1366 doi: 10.18653/v1/P19-1366