International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 5 Issue: 5 817 – 821

Generative Conversational Agents

The State-of-the-Art and the Future of Intelligent Conversational Systems

Arpan Bhattacharya
Empathize, India
e-mail: arpan82000@gmail.com

***Abstract* —** Intelligent conversational agents that generate responses from scratch are rapidly gaining in popularity. Sequence-to-sequence deep learning models are particularly well-suited for generating a textual response from a query. In this paper, I describe various generative models that are capable of having open-domain conversations. Toward the end, I present a null result I obtained in an attempt to train a chatbot from a small dataset and propose the use of a deep memory based machine translation model for training chatbots on small datasets.

Keywords- *chatbots; neural networks; machine learning; conversational models; deep learning; machine translation; artificial intelligence; statistics; computational linguistics; natural language processing; text mining*

*****

I. INTRODUCTION

2016 has been the year of the chatbot. Brands are increasingly employing chatbots to engage their customers. Various messaging platforms now host hundreds of chatbots that developers have built for them. Even though the chatbot space is getting more and more crowded, it's nowhere near saturated. Almost every phone app can be replaced by an intelligent chatbot.

There are two important aspects to a chatbot's functioning: decoding messages and generating responses. Different systems handle these aspects in different ways. Before deep learning hit the scene earlier this decade, all chatbots used to have hard-coded responses. These retrieval-based models didn't generate any new text; they just picked a response from a fixed set. Such models have been the focus of a number of works (Williams and Young, 2007 [1]; Schatzmann et al., 2007 [2][3]; Misu et al., 2012 [4]; Litman et al., 2000 [5]). However, retrieval-based models are unsuitable for long or open-domain conversations for the following reasons:

- These models lack flexibility. The responses need to be predefined and are difficult to customize for different situations or requirements.
- Since the responses need to be defined manually, the domain or scope of the conversations these models are capable of having is extremely narrow.

Generative models do not use predefined responses. They generate new responses from scratch. Generative models perform better than retrieval-based models as they can handle unseen cases and are often context-aware.

One of the metrics that can be used to assess the performance of most conversational models is the BLEU score. The algorithm was originally created for evaluating the quality of machine translated text. BLEU scores are known to reasonably correlate with human judgment. However, another metric introduced specifically for the assessment of generative conversational models, ΔBLEU, outperforms BLEU [6].

In the remainder of this paper, I summarize and comment on few of the most influential works in this direction that define the current state-of-the-art.

II. MODELS CAPABLE OF HAVING SHORT-TEXT CONVERSATIONS

In short-text conversations, the goal is to generate a single response to a single query. Short-text conversations are significantly easier to automate than long conversations. Measuring their performance is also relatively straightforward.

Most of the prominent works rely on massive amounts of conversational training data to scale to larger domains, where conversations can be on just about anything. A lot of work has also been done on producing natural responses in close-domain systems (Ratnaparkhi, 2000 [7]; Rambow et al., 2001 [8]).

Modern day generative models capable of having short-text conversations fall into two main categories:

1. SMT-based systems: Langner et al., 2010 [9] has covered the use of statistical machine translation (SMT) in translating internal dialogue state into natural language. Barzilay and Lapata, 2005 [10] also considers the user's utterance when generating responses in order to generate context-aware discourse. Galley et al., 2001 [11], Knight and Hatzivassiloglou, 1995 [12] and other works use information in the user's utterance to fill in knowledge gaps that might exist in the system. The best SMT-based model till date has been covered in Ritter et al, 2011 [13].
2. Neural Conversational Models: These models use neural networks for the task of response generation.

Most of them use two different networks: one that encodes the message into a hidden representation vector, and another that generates a response from the model, usually by multiplying it by a matrix that gets its values during training.

A. *Data-Driven Response Generation in Social Media (Ritters et al., 2011 [13])*

This paper presents a data-driven approach to generating responses to Twitter status posts. The conversational model discussed in this paper views response generation as a machine translation problem. Instead of translating from one language to another, they translate from a message (stimulus) to a response. This paper merits discussion as it went on to influence a number of works.

The fundamental assumption the authors of this paper make is that responses are always semantically parallel to the messages (Hobbs 1985 [14]).

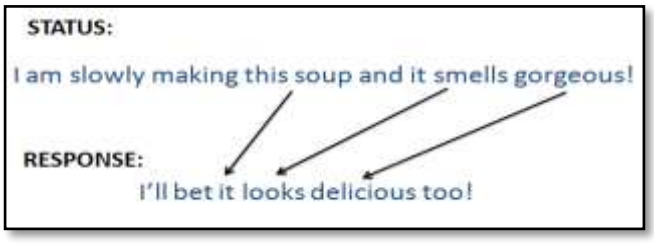

Figure 1. Semantic relationships between phrases in the status and the expected response

For instance, "it" in the above response refers to "this soup" in the status. There are similar relationships between "smells" and "looks", as well as "gorgeous" and "delicious". The conversational model described in this paper attempts to 'translate' a status to a response.

However, the problem of response generation differs from a typical machine translation problem in that the stimulus and the response are *not* semantically equivalent and hence, no intelligent system will ever be able to learn the semantics behind conversations. It is possible, however, for a system to learn some high-frequency phrase patterns that occur in conversations. For example, a response to a status that contains the phrase 'I am' will likely have the phrase 'you are' in it.

Recognizing such high-frequency phrase pairs in a dataset is the first step toward learning conversational models from data. But because the responses are not semantically parallel and/or equivalent, a lot of unaligned words were being initially left unprocessed. The researchers resolved this issue by generating all possible pairs of phrases that contain less than four words, and applying an association-based filter.

Using phrase-based statistical machine translation enabled the authors to leverage already existing techniques that are known to be accurate and scalable. SMT also provides a probabilistic model of responses, making it easy to integrate into production code.

This model outperformed retrieval based solution and its responses turned out to be better than actual human responses 15% of the time.

B. *Neural Responding Machine for Short-Text Conversation (Lifeng Shang, Zhengdong Lu, Hang Li 2015 [15])*

Although the work of Ritter et al [13] was novel and performed well on certain datasets, statistical machine translation seems to be intrinsically unsuitable for response generation due to the semantic inequivalence of the stimulus and the response. In this work, the authors use a neural encoder-decoder for the task of response generation, and also establish a probabilistic model that estimates the likelihood of a response given a post.

They used recurrent neural networks for both the encoder and decoder, because they're good at generating or analysing word sequences of variable lengths (Mikolov et al., 2010 [16]; Sutskever et al., 2014 [17] ; Cho et al., 2014 [18]).

The following steps describe the workflow of the conversational model the authors presented:

- The encoder converts the input sequence $x = (x_1, \cdots, x_T)$ into a set of hidden representations $h = (h_1, \cdots, h_T)$.
- These representations and the attention signal at time *t* are fed into the context generator.
- This forms the context input for the decoder network at time *t*: c_t.
- The decoder multiples the context input vector c_t by a matrix L, which gets its values during training, to get the t^{th} word of the response.
- The attention signal points to the part of the hidden response that needs emphasis.

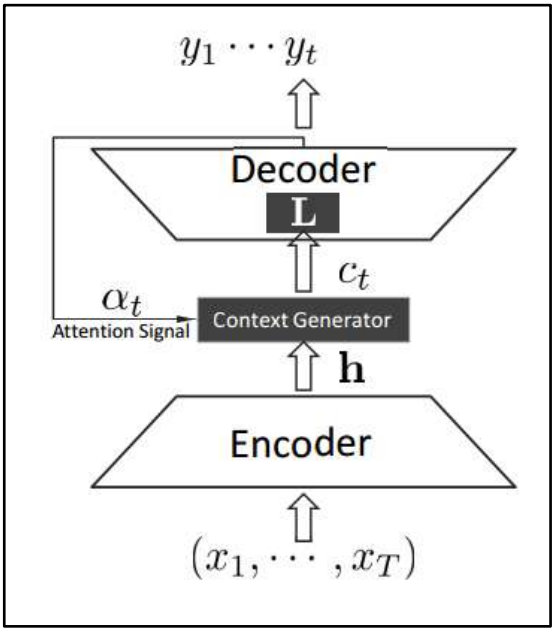

Figure 2. Encoder-decoder architecture of a Neural Responding Model

This work combines two recently proposed approaches: attention mechanism and encoder-decoder architecture. They train these models separately and then fine-tune the final combined model.

The nature of the model made it hard to apply BLEU scores, so they had to rely on human evaluation, and it happened to perform surprisingly better than retrieval- and translation-based models.

III. Models Capable Of Having Long Conversations

In long conversations, the chatbot goes through multiple rounds, and needs to remember the context. The following factors make building models capable of having long conversations a significantly harder challenge:

1. Context Sensitivity: The models may need to pick clues or information from previous dialogues. Thus, there needs to be a mechanism to preserve context information in the short term and the long term.
2. Consistency in Responses: A conversational agent needs to produce consistent responses to semantically equivalent queries. Incorporating a fixed personality into models is one of the toughest challenges in the field.

A. A Neural Conversational Model, Oriol Vinyals, Quoc Le [23]

This is the latest and most influential work in this area. In order to understand how this state-of-the-art model works, a thorough understanding of the underlying architectures that make it possible is needed.

1. Recurrent Neural Networks (RNN): Humans don't think from scratch when they have conversations. Our understanding of a word in a sentence depends on our understanding of the words and sentences preceding it. We use the information we learn from one part of a conversation to better our understanding of a later part of the conversation. This is where traditional neural networks fail us. A conversation model that uses (a pair of) traditional neural networks can't utilize the information in the previous queries it received and the responses it generated to answer future queries. RNNs contain cycles, thus allowing information to stay locked in it. This makes RNNs the ideal tools for learning to understand or generate sequences of text.
2. Long Short-Term Memory (LSTM) Networks: One of the drawbacks of naive RNN-based conversational models is that these models can't, in practice, learn from its past conversations unless they're very recent. LSTMs are hybrid RNNs capable of learning such long-term dependencies. They were introduced in a paper by Hochreiter et al (1997) [19]. LSTMs can 'remember' information for extended periods of time.

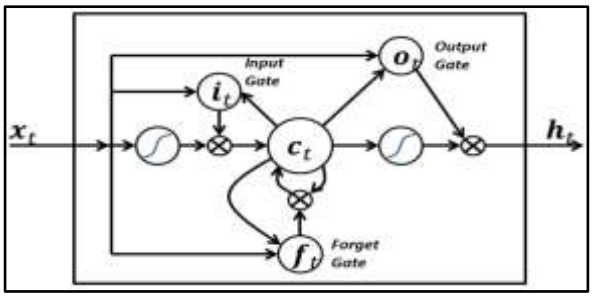

Figure 3. Four layers of an LSTM cycle
(Source: Wikipedia entry for LSTM)

The cycle in an LSTM network has four layers, unlike naive RNNs, whose cycles have just one *tanh* layer. It is important to note that information can easily flow through the line that passes through the output gate in the diagram unchanged or with only minor linear transformations. It's also worth noting that LSTMs don't use an activation function, so information is not dumbed down as it iterates through the layers.

3. Sequence-to-Sequence Learning: A paper by Ilya Sutskever et al (2015) [17] presented an end-to-end approach to learning to predict and/or analyze sequences of variable length using a multi-layered LSTM network. This approach turned out to be especially useful when the input and output are not necessarily equal in length or semantically equivalent, like in machine translation, This problem had also been attempted by Kalchbrenner and Blunsom [20], Cho et al, Graves [21], and Bahdanau et al [22]. However, Sutskever's work was an improvement over all of these. It got a BLEU score 34.81 in the WMT '14 English-French translation task.

This conversational model consists of just one multi-layer LSTM network that reads in a message one token at a time, and predicts the output sequence, one token at a time. During the training phase, the model is fed the correct output sequence, so that it's backpropagated to the source node. The model is trained to produce a higher value of cross-entropy for a better response given the context.

A TensorFlow implementation of this model, trained on the Cornell Movie Dialog Corpus, performed considerably better when several tokens from the previous step were also fed to the next step. The responses got slightly better when I used the probability of a past word appearing next to the predicted word to select the set of words from the previous step that were to be fed to the network.

Because of its simplicity, this model can be used not only for response generation, but also for machine translation and speech recognition.

IV. Other Significant Works

1. Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models (Yoshua Bengio et al 2016) [24]: Instead of LSTMs, this model builds on the hierarchical encoder-decoder model described in Sordoni et al. (2015a) where they used it for web search recommendations.
2. How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation (Chia-Wei Liu et al 2017) [25]: This study uncovers how every metric used to assess the performance of conversational agents is flawed, and how its results don't correlate with human evaluation. They were particularly successful in showing that the metrics like BLEU, which were meant for the assessment of machine translation systems, don't provide much useful feedback for response generation systems because the input and output are semantically dissimilar.

V. DISCUSSION

As of now, chatbots deployments are successful only when the domain of the conversations is extremely narrow and/or massive amounts of data is available to the developers. However, for a lot of use cases like automated psychotherapy, not a lot of data is available to researchers, mostly due to privacy concerns. Since every state-of-the-art conversational model relies on a huge amount of data, building one's own dataset is not a feasible option either. So basically, we need chatbots that need less data to train.

Here is a brief description of my attempt to train a therapist chatbot from a thread of just five-hundred text messages, which failed to perform reasonably well.

I implemented the model described in [23] in TensorFlow, but I used a neural turing machine [26] instead of an LSTM network. By combining the recurrent neural network with external memory resources, the system can make slow changes in its parameters in the long term, and leverage the power of internal memory to make some immediate changes during training. This feature is important because since we lack training data, the model will need to make some quick and drastic changes to the parameters, while avoiding completely relearning the parameters every time new data is fed. Though, upon human evaluation, the performance of this model seems to be far from decent, an important observation is that the performance went up on training the model with the same data multiple times. This observation is in agreement with the claim in [27].

There's a chance the deep-memory based machine translation model discussed in [28] can be used to solve this problem. The method they describe performed as well as the statistical phrase-based machine translation system Moses with a considerably smaller vocabulary and parameter-size. Although this is mere speculation, if one manages to implement this sequence-to-sequence learning model for chatbots, it will probably perform relatively well, even with minimal training.

VI. Conclusion

I have presented a comprehensive summary of the most important papers on intelligent conversational systems, and discussed the need for conversational models that need less data to train on. I've also proposed that memory-augmented neural systems be used for building chatbots that can be trained using less data.

Acknowledgment

I would like to thank Mr Debarghya Das, Software Engineer at Google NYC working on knowledge graph, query understanding and personalization for Search, Mr Raziman Thottungal Valapu, nanophotonics postdoc at École Polytechnique Fédérale de Lausanne, Switzerland, Mr Ankit Shetty, Computer Science undergrad student at Indian Institute of Technology, Bombay, Ms Harshita Ghirase, Computer Science undergrad student at the State University of New York at Buffalo. and Mr Harsh Goyal, Computer Science undergrad student at the University of Texas at Austin, for their comments and feedback on an earlier version of this manuscript, although any errors are my own and should not tarnish the reputations of these esteemed persons..

References

[1] Jason D Williams and Steve Young. 2007. Partially observable markov decision processes for spoken dialog systems. Computer Speech & Language 21(2):393–422.

[2] Schatzmann, J., Thomson, B., Weilhammer, K., Ye, H., and Young, S. (2007a). Agenda-based user simulation for bootstrapping a pomdp dialogue system. In Proc. of the North American Meeting of the Association of Computational Linguistics (NAACL).

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] Teruhisa Misu, Kallirroi Georgila, Anton Leuski, and David Traum. 2012. Reinforcement learning of question-answering dialogue policies for virtual museum guides. In SIGDIAL, pages 84–93. ACL.

[5] Diane Litman, Satinder Singh, Michael Kearns, and Marilyn Walker. 2000. Njfun: a reinforcement learning spoken dialogue system. In Proceedings of the 2000 ANLP/NAACL Workshop on Conversational systems, pages 17–20. ACL.

[6] ΔBLEU: A Discriminative Metric for Generation Tasks with Intrinsically Diverse Targets Michel Galley Chris Brockett Alessandro Sordoni Yangfeng Ji, Michael Auli Chris Quirk Margaret Mitchell1 Jianfeng Gao1 Bill Dolan

[7] Adwait Ratnaparkhi. 2000. Trainable methods for surface

natural language generation. In Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference

[8] Owen Rambow, Srinivas Bangalore, and Marilyn Walker. 2001. Natural language generation in dialog systems. In Proceedings of the first international conference on Human language technology research, HLT '01, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

[9] Brian Langner, Stephan Vogel, and Alan W. Black. 2010. Evaluating a dialog language generation system: comparing the mountain system to other nlg approaches. In INTERSPEECH

[10] Regina Barzilay and Mirella Lapata. 2005. Modeling local coherence: an entity-based approach. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05.

[11] Michel Galley, Eric Fosler-Lussier, and Alexandros Potamianos. 2001. Hybrid natural language generation for spoken dialogue systems. In Proceedings of the 7th European Conference on Speech Communication and Technology (EUROSPEECH–01), pages 1735–1738, Aalborg, Denmark, September.

[12] Kevin Knight and Vasileios Hatzivassiloglou. 1995. Two-level, many-paths generation. In Proceedings of the 33rd annual meeting on Association for Computational Linguistics, ACL '95.

[13] Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In EMNLP, pages 583–593. Association for Computational Linguistics.

[14] Jerry R. Hobbs. 1985. On the coherence and structure of discourse.

[15] Shang, L., Lu, Z., and Li, H. Neural responding machine for short-text conversation. In Proceedings of ACL, 2015.

[16] Tomas Mikolov, Martin Karafiat, Lukas Burget, Jan Cernocky, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In INTERSPEECH 2010, pages 1045–1048.

[17] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. 2014.Sequence to sequence learning with neural networks. In NIPS, pages 3104–3112.

[18] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

[19] Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long short-term memory. Neural computation, 9(8):1735–1780.

[20] Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In EMNLP, pages 1700–1709.

[21] Alex Graves. 2013. Generating sequences with recurrent neural networks. preprint arXiv:1308.0850.

[22] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

[23] Oriol Vinyals and Quoc Le. 2015. A neural conversational model. In Proc. of ICML Deep Learning Workshop.

[24] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2015. Building end-to-end dialogue systems using generative hierarchical neural network models. In Proc. of AAAI

[25] How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation. Chia-Wei Liu, Ryan Lowe, Iulian V. Serban, Michael Noseworthy, Laurent Charlin, Joelle Pineau. arXiv preprint arXiv:1603.08023v2

[26] Graves, Alex, Wayne, Greg, and Danihelka, Ivo. Neural turing machines. arXiv preprint arXiv:1410.5401, 2014.

[27] One-shot Learning with Memory-Augmented Neural Networks. Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, Timothy Lillicrap. Google DeepMind, 2016

[28] .Fandong Meng, Zhengdong Lu, Zhaopeng Tu, Hang Li, and Qun Liu. 2015. A deep memory-based architecture for sequence-to-sequence learning. In Proceedings of ICLR-Workshop 2016, San Juan, Puerto Rico.