Project Proposal for Domain Adaptation in NLG

Benny Longwill

November 2021

1 Introduction

Automated natural language generation (NLG) stands at the front line of human-computer interaction by allowing computers to speak our language. Analogous to how humans convert thoughts into writing, NLG allows computers to translate data into human understandable speech or text. In the same way humans communicate with other humans, computers could streamline the benefits of many applications like automatic summarization, question answering and language translation. A breakthrough in this domain could cause a cascading effect for research into applications in which NLG plays an integral role.

One problem currently under investigation is the ability for dialogue systems to adapt to their human interlocutors. (Janarthanam and Lemon, 2014) Because current systems are widely unable to perceive individual variation in language comprehension of the user, the utilization of overly complex or simple language could cause misunderstandings or unnaturalness in the dialogue. (Isaacs and Clark, 1987) For this reason, there is a call for research toward individualized language interaction between humans and their computer systems.

Recently, neural language model research has received state-of-the-art results in NLG and are often a top choice for system architecture. Massively pre-trained transformer models like GPT-2 and GPT-3 are able to produce generated text that is at times indistinguishable from human-written text.(Bahdanau et al., 2015) Despite this success, using parametric-only NLG in a conversational agent application that directly interact with users is still problematic due to unreliability in the generated text.

Although pre-trained neural-NLG models may seem coherent and informative due to the parametric knowledge encoded by their weights, they are also known to produce text that is random and off-topic.(Dale, 2021; Marcus, 2020; Clark et al., 2018) For this reason there is still a large demand for more human-like, relevant, and well-informed dialog generation in end-to-end systems. (Li et al., 2016; Zhang et al., 2018; Shuster et al., 2020) Recently there has been successful work by Lewis et al. (2020) on improving a pre-trained model's ability to access and employ knowledge for improved performance on downstream knowledge-intensive NLP tasks. Retrieval-Augmented Generation (RAG) provides a robust question answering (QA) framework in which a relevant text from corpora is retrieved and encoded such that generated dialogue can be conditioned on non-parameterized knowledge in addition to the user query. This method of generation was found to be more specific, diverse, and factual than state-of-the-art (SOTA) parametric-only baselines. In spite of this success, the Seq2Seq model used as the generator base in RAG could still be improved through the implementation of controllable generation.

Currently the necessity for style-control in NLG is high for abstractive generation tasks (See et al., 2019) like open-ended QA systems. Because different use cases for such a system also imply different model behavior unique to the domain, there is high demand for methods that dynamically change style attributes of generated text while holding content the same. For example, in the same vein as a customer service chat-bot that requires text generation with a positive sentiment attribute, a RAG QA system could also use style transfer to tailor its generated explanations or summaries to

accommodate a user's level of comprehension. Attributes like specificity, vocabulary, and response-relatedness can be modified to match the user whether it be for a student, grandmother, or expert in the field. As such, investigation into the application of style-transfer to text may indicate its efficacy as a tool for individualized interaction in QA dialogue systems.

Previous studies have reported distinct procedures for style-control on text generation with some small similarities. For example, Li et al. (2018) proposed a method for text attribute transfer in which style is stripped from source text using a word frequency heuristic, new sequences of similar content are retrieved from a corpus of a different style, and an autoregressive (AR) neural model combines the two into a fluent style-transferred sequence. Results showed that the procedure outperformed previous SOTA adversarial-based models by 22% in human evaluation of grammar and appropriateness of sentiment-transferred responses. Furthermore, Dathathri et al. (2019) also leveraged AR generation and word frequency. They guide generation by applying a simple attribute classifier (e.g., bag-of-words model) to a pre-trained AR generator. At each time step, there is a forward and backward pass of the generator context through the attribute classifier to calculate gradients that are then applied to the generator's hidden activations. In this way, the desired style is surfaced when decoding the generators output. Results showed that PPLM outperformed weighted decoding (WD) (Holtzman et al., 2018; Ghazvininejad et al., 2017) and CTRL (Keskar et al., 2019) in automated evaluation and are comparably fluent despite their size. It is discussed that PPLMs could be used with any attribute model architecture, but neither Dathathri et al. (2019) nor Li et al. (2018) investigate a style-transfer procedure that uses non-autoregressive (NAR) generation. Thus, the affect of style-transfer on NAR generation is unclear.

The benefit of NAR generation is that it outperforms autoregressive (AR) generation in terms of much quicker inference speed - a welcomed attribute for RAG QA during training (Lewis et al., 2020). Because AR models like RAG's Seq2Seq generate sequences from left-to-right and make token predictions one position at a time by conditioning on the previous context, the NLG associated with vanilla RAG is innately non-parallelizable and has high computational overhead and latency. For this reason, an approach that incorporates style-transfer into a framework optimized with NAR efficiency would also be of great value toward individualized interaction. One such NLG decoding strategy that could lend itself well toward the adoption of style-transfer is *Mask-Predict* from Ghazvininejad et al. (2019). In *Mask-Predict*, a NAR model predicts all positions of output from one pass and improves the generation over a constant number of time steps by iteratively re-generating subsets of tokens for which the model has low confidence. Not only does this method come near AR models in terms of performance, but the re-iterative prediction strategy can easily re-oriented toward generating text of another style.

The purpose of the current study is to investigate the effects of a novel style-transfer method for text through the implementation of a dialogue system that tailors discourse to the level of the user. Based on the robustness and stability demonstrated by Zeng et al. (2020), a form of the RAG question and answering framework will be implemented as the base of the system. In this way, RAG will determine the content of what is being generated. Additionally, its NLG module will be augmented with style-transfer in order to generate text fitted to the user.

To this end, a Naive-Bayes (NB) language classification model (McCallum et al., 1998) will be implemented based on the results of the style detection heuristic from Li et al. (2018) and the BOW model from Dathathri et al. (2019). This is due to the simplicity and effectiveness in using probabilistic models. Implementing a lightweight model that has higher interpretability may increase efficiency and reduce the computational cost of training larger neural models. As such, the NB model will play a multifaceted role in the NLG module.

Although NB has strong independence assumptions, it stands as a pragmatic baseline for NLP tasks like sentiment classification. (Wang and Manning, 2012) For this reason, the NB model will serve to monitor the human user input signal, and to make predictions about the level of user language comprehension. These prediction will indicate to the NLG module which style should be used to generate responses.

Based on the results of Dathathri et al. (2019), this NB classifier will also play the role of an attribute model from *PPLM*. Because the model will already be required to learn a representation of token class salience for its user classification task, it it will further cut down on the cost of computational resources to re-use the same model for directing the generator. Additionally, knowing more about how to effectively control fine-grained features (e.g., wordiness, vocabulary level, or coherence relations) in generated text would lead to more clear and comfortable communication between dialogue systems and their human users.

Finally, due to the results of Ghazvininejad et al. (2019) and the inductive bias that style is localized in discrete phrases from Li et al. (2018), the *Mask-Predict* decoding strategy will be utilized to generate tokens from a NAR generator. In this way, the NB classifier will be used to indicate to *Mask-Predict* where style is needed in generated phrases. Exploration into how to efficiently generate fluent text using a NAR generator would speed up inference in a wide variety of NLP applications including, machine translation, speech recognition, and text to speech. (Ren et al., 2020)

2 Proposed Method

2.1 Datasets

2.1.1 Non-Parametric Knowledge

Although Lewis et al. (2020) considered a wide array of evaluation throughout different experiments, the primary goal of the current study is abstractive question answering. As such, any internet wiki (e.g., Wookiepedia, Game of Thrones wiki, Wikipedia dump) could be scraped or downloaded to provide the non-parametric knowledge in the DPR module.

2.1.2 Question Answering

Also in following Lewis et al. (2020), the MSMARCO is a human made machine reading comprehension dataset (Bajaj et al., 2018) that provides queries and gold standard responses for training the system end-to-end. Queries were extracted from a Bing search engine document, and the answers to the queries were human generated. Another consideration to fill this same role is the Stanford QA dataset (SQUAD) I and II. (Rajpurkar et al., 2016) This dataset consists of 100,000+ questions and answers posed by crowdsourced workers about Wikipidia articles. Spans for correct answers from the corresponding passages are also provided.

2.1.3 Attribute Model

Wikipedia may also be useful for training the attribute model on different level of language comprehension. As the example presented in Figure 1 demonstrates, standard Wikipedia articles are often written in the technical style of an encyclopedia. For this reason, authors also separately provide articles of the same topic, like the one presented in Figure 2, that utilize simplified English in order to make the knowledge more accessible to children or adult second language learners. This form of English demonstrates a more basic vocabulary, grammar, and shorter sentences. Because these simplified English articles are not necessarily shorter in length, they would also be a good candidate to provide a contrasting register for the proposed NLG system.

Another consideration for training the attribute model get obtain a sense of comprehension level is the corgis dataset based on the top most popular books on project Gutenberg classics novels. (Bart et al., 2017) The dataset provides different readability measures including the Fleish-Kincaid index and grade level.

Chemical engineering is a branch of engineering which deals with the study of design and operation of chemical plants as well as methods of improving production. Chemical engineers develop economical commercial processes to convert raw material into useful products. Chemical engineering uses principles of chemistry, physics, mathematics, biology, and economics to efficiently use, produce, design, transport and transform energy and materials. The work of chemical engineers can range from the utilization of nanotechnology and nanomaterials in the laboratory to large-scale industrial processes that convert chemicals, raw materials, living cells, microorganisms, and energy into useful forms and products.

Chemical engineers are involved in many aspects of plant design and operation, including safety and hazard assessments, process design and analysis, modeling, control engineering, chemical reaction engineering, nuclear engineering, biological engineering, construction specification, and operating instructions.

Chemical engineers typically hold a degree in Chemical Engineering or Process Engineering. Practicing engineers may have professional certification and be accredited members of a professional body. Such bodies include the Institution of Chemical Engineers (IChemE) or the American Institute of Chemical Engineers (AIChE). A degree in chemical engineering is directly linked with all of the other engineering disciplines, to various extents.

Figure 1: Example of an article written in Standard Wikipedia English
t

Chemical engineering is a branch of engineering dealing with chemistry that came to existence in the early 20th century.[1] Before this time, chemical plants were designed by chemists, who were trained to work on a small scale only. Chemical engineering combines the jobs of a chemist and that of industrial engineer. This makes factories more efficient and chemicals much cheaper. Chemical engineering uses physics (the science of moving objects and forces), chemistry (the science of substances), and mathematics. There are many different types of jobs for people with degrees in chemical engineering.

Some new topics in chemical engineering include:

- environmental sanitation at factories (making certain that nature is not hurt by the factories);
- developing types of energy other than those from gas or oil;
- biomedical engineering.

Figure 2: Example of an article written in simplified English

2.2 Architecture

Figure 3 presents the proposed architecture for the current study. This architecture contains a Dense Passage Retriever, Non-Autoregressive BART generator and a light weight Naive Bayes Attribute model.

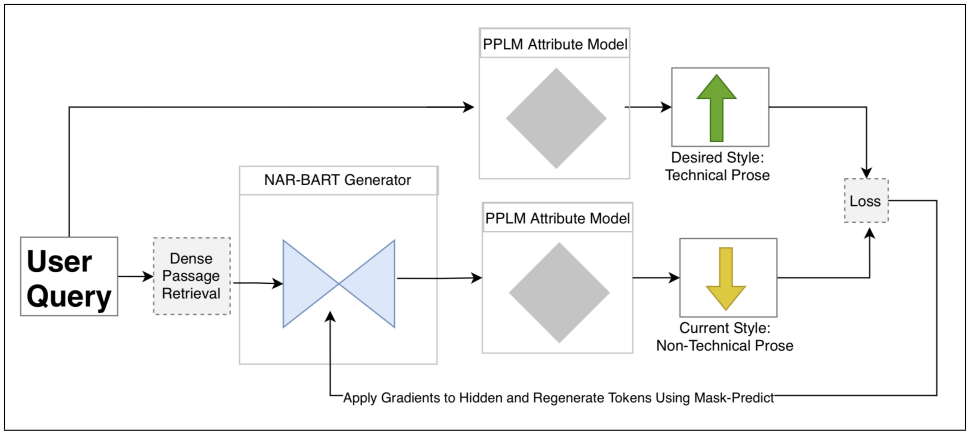

Figure 3: Proposed System Architecture

2.3 Procedure

The system begins when the user makes a query. As shown in Figure 4 DPR is used to encode the query and find the top most relevant documents based on a cosin similarity. The retrieved documents are concatenated together with the user query and input into the generator. In following the generation algorithm described below, the system response is created and output.

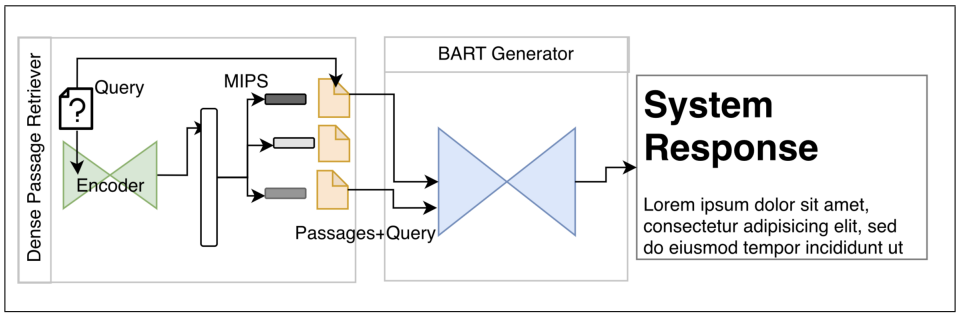

Figure 4: The Retrieval-Augmented Generation framework as it is depicted in Lewis et al. (2020) A pre-trained retriever is paired with a pre-trained seq2seq generator. The maximum inner product (MIPS) is used to locate the top most relevant documents form the non-parametric knowledge base. Retrieved documents are concatenated with the user query and marginalized by the generator in order to generate a system response.

2.3.1 User-Query Classification and Attribute Model

User queries are tokenized and classified according to a multi-class Naive Bayes classifier. Features are the real value number of token occurences. Laplace smoothing is utilized to obtain obtain a smooth representation by adding 1 to all n-gram counts before normalization.

The following is the formula used for classification:

$$classify(d_i) = argmax_c P(c) \prod_{k=1}^{|V|} P(w_k|c)^{N_{ik}}$$

where class c prior probability is :

$$P(c_i) = \frac{1 + Cnt(c_i)}{|C| + \sum_i Cnt(c_i)}$$

and the word w conditional probability is:

$$P(w_t|c_j) = \frac{1 + \sum_{i=1}^{|D|} N_{it} P(c_j|d_i)}{|V| + \sum_{s=1}^{|V|} \sum_{i=1}^{|D|} N_{is} P(c_j|d_i)}$$

where V represents the vocabulary set and D represents document set

2.3.2 RAG Retrieval

The retriever system for RAG uses a neural method called dense passage retrieval (DPR). DPR consists of two neural models and a dataset of questions paired with relevent passages. For training, one model encodes the questions and the other encodes the passages. A similarity score is then derived for encoded questions paired with encoded passages. A loss function compares the similarity

score with the gold standard label. The loss is backpropogated through both models. As such, trained DPR has adjusted embeddings in order to have a better ability to determine whether or not two pieces of text are similar. At inference time, DPR recieves a single query and calculates the similarity between this query embedding and all embedded passages. The passages with the highest similarity are returned.

2.3.3 Generation

In following a *Mask-Predict* procedure augmented by *PPLM*, a [MASK] token vector is input to the NAR generator in order to predict an entire sequence in parallel. This generated sequence is then input to the attribute model and feature weights are derived for each token. Based on these weights, a class prediction (e.g., low comprehension level, high comprehension level) is made. A loss is then calculated using a Maximum Entropy function that compares similarity between the sample prediction and the current user comprehension level. A backward pass is made and the gradients are generated and saved.

In moving into a subsequent iteration of *Mask-Predict*, a subset of tokens are selected based on the attribute-model's feature weights for the each of the tokens in the currently desired class style. As such, tokens with log probability scores below a threshold are replaced with a [MASK] token. This new sequence is then passed through the model again to re-sample the masked tokens in parallel while conditioning on the new bidirectional context. In order to provoke the desired style into generation, the gradients from the attribute model are directly applied to the model's hidden state. This process then repeats itself over ten time steps.

2.3.4 Evaluation

The current study will closely follow Hinton et al. (2015) in evaluation for comparison. Specifically the procedure using the MSMarco NLG v2.1 (Bajaj et al., 2018) abstractive QA generation task will be implemented. In this task, the system is queried using questions from the dataset. Responses are generated in conjunction with a wikipedia dump as a non-parametric knowledge base. ROUGE-L and BLEU between generated responses and gold-standard are reported.

There are also several measures to evaluate the general NLG fluency and diversity of the generator, used in previous literature, that would also prove useful in the current study: Corpus-BLEU, ROUGE, Self-BLEU, Perplexity, and Human Evaluation. In following Hinton et al. (2015) metrics should confirm fluency, make use of the specified attribute values in generations, and preserve the rest of the content of the input. In the style of 'A-B' testing, scores would be used to compare 1000 samples from a pre-trained generator out-of-the-box in with a specified number of samples of the same generator after it has been fine-tuned. Aside from fluency, a style score should also be included in order to confirm whether the generation has the target style.

2.3.5 Corpus-BLEU

BLEU scores compare n-gram overlap between a generated text sample and a corresponding reference text as a measure of similarity. In the current study, the original corpus on which the system was trained could be used as a reference in order to see how generated text is affected by the fine-tuning. Given that the original data was human derived and published on Wikipedia, it is assumed that the dataset text is also well formed and coherent. Therefore, higher BLEU between the original data and generated text would signify greater fluency in the generated text.

2.4 ROUGE

Specifically Hinton et al. (2015) utilized ROUGE in their abstractive QA generation task. ROUGE is highly similar to BLEU except that it measures how much the n-grams in the gold standard

reference appeared in the generated text. Rather than the other way around. Naturally ROUGE complements BLEU often times parallel to the relationship that recall has with precision. (Lin, 2004)

2.4.1 Self-BLEU

Self-BLEU compares a single generated sample in reference to all the remaining generated samples within a batch. As opposed to traditional BLEU, a high score would indicate low diversity among samples and could indicate the presence of mode collapse. In contrast, a lower average Self-BLEU score would signify greater uniqueness among each sample, demonstrating greater variation in the output.

2.4.2 Perplexity

Perplexity is a statistical metric for evaluating the ability of a language model to predict the testing data. In order to calculate this score, an outside language model trained on the same dataset would be required to provide this measure. Lower perplexity scores signify that the data distribution of the model makes good predictions when sampling.

2.4.3 Human Evaluation

Human evaluation is often considered the gold standard in NLG. Although it is costly, flaws in the aforementioned automated evaluation methods could outweigh the time and effort cost involved with having humans grade the fluency and coherence of generated text. In addition to creating surveys for use in the local context, online crowd sourcing like Amazon Mechanical Turk is a strong online platform that could further facilitate human judgement for this system.

2.4.4 Style Score

An outside classifier could be trained to report the style of generations in comparison to their target. This measure would be used to confirm that generations are in fact meeting the feature requirements of their style category.

References

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1409.0473`.

P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, and T. Wang. Ms marco: A human generated machine reading comprehension dataset, 2018.

A. C. Bart, R. Whitcomb, D. Kafura, C. A. Shaffer, and E. Tilevich. Computing with corgis: Diverse, real-world datasets for introductory computing. *ACM Inroads*, 8(2):66–72, mar 2017. ISSN 2153-2184. doi: 10.1145/3095781.3017708. URL `https://doi.org/10.1145/3095781.3017708`.

E. Clark, Y. Ji, and N. A. Smith. Neural text generation in stories using entity representations as context. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2250–2260, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1204. URL `https://aclanthology.org/N18-1204`.

K. Dabas, N. Madaan, V. Arya, S. Mehta, G. Singh, and T. Chakraborty. Fair transfer of multiple style attributes in text. *CoRR*, abs/2001.06693, 2020. URL https://arxiv.org/abs/2001.06693.

R. Dale. Gpt-3: What's it good for? *Natural Language Engineering*, 27(1):113–118, 2021. doi: 10.1017/S1351324920000601.

S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, and R. Liu. Plug and play language models: A simple approach to controlled text generation. *CoRR*, abs/1912.02164, 2019. URL http://arxiv.org/abs/1912.02164.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

M. Ghazvininejad, X. Shi, J. Priyadarshi, and K. Knight. Hafez: an interactive poetry generation system. In *Proceedings of ACL 2017, System Demonstrations*, pages 43–48, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL https://aclanthology.org/P17-4008.

M. Ghazvininejad, O. Levy, Y. Liu, and L. Zettlemoyer. Constant-time machine translation with conditional masked language models. *CoRR*, abs/1904.09324, 2019. URL http://arxiv.org/abs/1904.09324.

H. Gong, S. Bhat, L. Wu, J. Xiong, and W. W. Hwu. Reinforcement learning based text style transfer without parallel training corpus. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 3168–3180. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1320. URL https://doi.org/10.18653/v1/n19-1320.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014. URL http://papers.nips.cc/paper/5423-generative-adversarial-nets.

G. E. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015. URL http://arxiv.org/abs/1503.02531.

A. Holtzman, J. Buys, M. Forbes, A. Bosselut, D. Golub, and Y. Choi. Learning to write with cooperative discriminators. *CoRR*, abs/1805.06087, 2018. URL http://arxiv.org/abs/1805.06087.

Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing. Toward controlled generation of text. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1587–1596. PMLR, 2017. URL http://proceedings.mlr.press/v70/hu17e.html.

E. Isaacs and H. Clark. References in conversation between experts and novices. *Journal of Experimental Psychology General*, 116:26–37, 03 1987. doi: 10.1037/0096-3445.116.1.26.

S. Janarthanam and O. Lemon. Adaptive generation in dialogue systems using dynamic user modeling. *Computational Linguistics*, 40(4):883–920, Dec. 2014. doi: 10.1162/COLI_a_00203. URL https://aclanthology.org/J14-4006.

E. Jang, S. Gu, and B. Poole. Categorical reparameterization with gumbel-softmax. *CoRR*, abs/1611.01144, 2016. URL http://arxiv.org/abs/1611.01144.

N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, and R. Socher. Ctrl: A conditional transformer language model for controllable generation, 2019.

P. S. H. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401, 2020. URL https://arxiv.org/abs/2005.11401.

D. Li, Y. Zhang, Z. Gan, Y. Cheng, C. Brockett, B. Dolan, and M. Sun. Domain adaptive text style transfer. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3302–3311. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1325. URL https://doi.org/10.18653/v1/D19-1325.

J. Li, M. Galley, C. Brockett, G. P. Spithourakis, J. Gao, and W. B. Dolan. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016. doi: 10.18653/v1/p16-1094. URL https://doi.org/10.18653/v1/p16-1094.

J. Li, R. Jia, H. He, and P. Liang. Delete, retrieve, generate: A simple approach to sentiment and style transfer, 2018.

C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

G. Marcus. The next decade in AI: four steps towards robust artificial intelligence. *CoRR*, abs/2002.06177, 2020. URL https://arxiv.org/abs/2002.06177.

A. McCallum, K. Nigam, et al. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer, 1998. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.65.9324&rep=rep1&type=pdf,/bib/mccallum/mccallum1998comparison/mccallum1998nbayes.pdf.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250, 2016. URL http://arxiv.org/abs/1606.05250.

Y. Ren, J. Liu, X. Tan, Z. Zhao, S. Zhao, and T.-Y. Liu. A study of non-autoregressive model for sequence generation, 2020.

A. See, S. Roller, D. Kiela, and J. Weston. What makes a good conversation? how controllable attributes affect human judgments. *CoRR*, abs/1902.08654, 2019. URL http://arxiv.org/abs/1902.08654.

K. Shuster, S. Humeau, A. Bordes, and J. Weston. Image-chat: Engaging grounded conversations. In D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2414–2429. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.219. URL https://doi.org/10.18653/v1/2020.acl-main.219.

Y. Su, D. Cai, Y. Wang, D. Vandyke, S. Baker, P. Li, and N. Collier. Non-autoregressive text generation with pre-trained language models, 2021.

S. Wang and C. Manning. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 90–94, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://aclanthology.org/P12-2018.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8:229–256, 1992. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

K. Zeng, M. Shoeybi, and M. Liu. Style example-guided text generation using generative adversarial transformers. *CoRR*, abs/2003.00674, 2020. URL https://arxiv.org/abs/2003.00674.

H. Zhang, J. Xu, and J. Wang. Pretraining-based natural language generation for text summarization, 2019.

S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In I. Gurevych and Y. Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2204–2213. Association for Computational Linguistics, 2018. doi: 10.18653/v1/P18-1205. URL https://www.aclweb.org/anthology/P18-1205/.