Answer-Agnostic Question Generation in Privacy Policy Domain using Sequence-to-Sequence and Transformer Models

Deepti Lamba
Department of Computer Science
Kansas State University
Manhattan, Kansas, U.S.A
Email: dlamba@ksu.edu

William H. Hsu
Department of Computer Science
Kansas State University
Manhattan, Kansas, U.S.A
Email: bhsu@ksu.edu

***Abstract*—This paper presents a transformer and sequence-to-sequence mapping approach, augmented using relevant named entities, towards generating questions for text understanding in the domain of policies (such as privacy agreements). To date, most work in question generation has used general-purpose benchmarks such as the SQuAD corpus; however, the application of question generation to document understanding has largely omitted use cases in the privacy policy domain. A privacy policy is a legal document that clearly divulges the policies of an organization regarding the gathering and usage of customer data. Reading and understanding privacy policies to gain a measure of the rationality of terms of service before accepting them is of critical importance to users. However, users frequently ignore these policies or base their decisions on insufficient understanding of the "fine print" due to the complex language and excessive length of policies. This work focuses on building a question generation system to assist reading comprehension of privacy policies. This work can be used to improve the quality of question-answering systems in this domain, which consequently can help users understand the privacy policies before agreeing to them. This paper uses existing deep learning models like T5, which is a transformer model, and several sequence-to-sequence models to generate questions. Since, existing named entity recognition (NER) tools do not work in this domain, we also created our own named entity labels to add auxiliary information to our models. Adding the auxiliary information improves the results over the baseline models giving us a promising future direction to further add some constraints, as a means to add more background knowledge to our models to generate better quality questions.**

***Index Terms*—Sequence-to-sequence models; Privacy policy; Question generation; Transformer; named entities;**

I. Introduction

As the name suggests, Question Generation (QG) is the task of formulating interrogative sentences for comprehending text from a given context. Rus et al. (2008) [1] defined the QG task as "the automatic generation of questions (Factual questions, Yes/No questions, Why-questions, etc.) from inputs such as text, raw data, and knowledge bases." Some well-known applications of question generation are in: intelligent tutoring systems in educational domain [2] [3]; dialogue systems such as chat bots to enhance human-machine interactions [4]; and QA systems to improve their performance. [5] [6] [7]. This work is focused on generating questions from privacy policies documents, which are legally binding agreements between a consumer and an organization that precisely state the data collecting, data handling, and data processing practices adopted by the organization.

In order to protect one's privacy, it is crucial that a user reads and understands the privacy policies of any service before agreeing to them. However, research has shown that users generally avoid reading these documents due to several factors, which include: the seemingly immoderate document length [10] [11] [12], the perceived intricacy of the document language [13] [12], and the lack of willingness on the part of humans to invest time in reading the documents in their entirety [10]. The objective of this research is to use privacy policy documents to create an automatic question generation system, which can be then be used to build a question answering system. A relevant use case of this work could involve generating questions and their responses to create a frequently asked questions section for a given application, thereby, allowing users to understand privacy policies in a comparatively easier manner. However, the development of such a system in the domain comes with its challenges, for instance, obtaining a well annotated, large size data set, where annotation is done by domain experts (e.g., law students, lawyers). To date, we have found only two such data sets that can be used for building question generation or question answering systems: Ravichander et al. (2019) [8] and Ahmad et al. (2020) [9]. Table I gives the details of the two data sets mentioned above.

In recent years, sequence-to-sequence models have been widely used for automatic question generation. However, recently question generation systems have been built using transformer-based models. Our goal here is to automatically generate questions in the privacy policy domain using existing models: T5 transformer model and sequence-to-sequence models. Additionally, a study of the privacy policy documents show that the existing named entity recognition (NER) tools are not sufficient to label a majority of the examples in our data set. As a result, we also define our own custom named entities in this domain to provide auxiliary information to the

TABLE I
PRIVACY POLICY DATA SETS FOR QUESTION-ANSWERING TASK

Author	Name	#Policies	#examples
Ravichander et al. (2019) [8]	PrivacyQA	35 mobile applications	3500
Ahmad et al. (2020) [9]	PolicyQA	115 websites	25017

model during training. To date, most of the work in the privacy policy domain has centered around knowledge extraction using question answering, topic-modeling, and rule-based systems. To our best knowledge, this paper marks the first attempt to build an automatic question generation model in the privacy policy domain.

The following points summarize the contributions of this paper:

- This research uses existing transformer-based and sequence-to-sequence models to generate questions in the privacy policy domain. To the best of our knowledge, this effort to generate questions is the first of its kind in the privacy policy domain.
- Additionally, this work also creates named entity labels for the domain because existing NER tools are inapt due to the nature of these documents.
- Providing named entity labels as input to sequence-to-sequence models produces an increase in performance over the baseline models. This is in line with the work of Zhou et al. (2017) [14]. Doing the same for the T5 model [15] also produced an improvement over the baseline model.

This serves as the first step in our work to provide a deep learning architecture for the privacy policy domain to generate syntactically and semantically correct questions. We conducted extensive experiments and present our results using BLEU-n [16], METEOR [17], and ROUGE-L [18]. We are currently improving this work with the use of constraints to provide additional background knowledge about the context during training.

II. RELATED WORK

We briefly survey and distinguish between rule-based and neural-based systems used for question generation in the following sub-section. This review also covers some of the latest work in question generation using transformer-based approaches.

A. *Rule-based Systems*

Rule-based systems use hand-crafted rules for question generation, which require extensive linguistic knowledge and time to build. However, these systems offer some advantages as they are interpretable, less data intensive, and provide more control to developers. Some prominent work in this area was provided by: Dhole and Manning (2020) [19], Khullar et al. (2018) [20], Chali and Hasan (2015) [21], Heilman and Smith (2009) [2], [22], Gates (2008) [23], and Mitkov and Ha (2003) [24].

B. *Neural Question Generation*

The earliest work to use a deep-learning approach to generate questions was presented by Du. et al (2017) [25], where a sequence-to-sequence model (Sutskever et al., 2014 [26]) was used with the attention mechanism (Bahdanau et al., 2014) [27]. A vast majority of the research that uses deep learning for question generation uses sequence-to-sequence models. Pan et al. (2019) [28] present a extensive survey on the latest advances in question generation research. Providing auxiliary information as input to the model, in addition to the context, is a recurring theme that was observed. Some prominent work that used auxiliary information are: Zhou et al. (2017) [14] where the answer position, part of speech (POS) tags (Brill, 1992 [29]) and NER based features were used; Ma et al. (2020) [30] used NER features, POS tags, word case, and answer positions; Harrison and Walker (2018) [31] used word case, named entity recognition, and entity co-reference resolution. Models that used answer or answer positions suffered from the issue of answer words being included in the generated questions, which lowered the quality of the generated questions. Kim et al. (2019) [32] provided a solution by masking the answer in the input. Another approach in neural question generation was encoding the input passage or context separately from the answer (Song et al. (2018) [33]). Harrison and Walker (2018) [31] used one encoder each for sentence level embedding and token level embedding.

This work uses sequence-to-sequence model with and without attention, but instead of using linguistic features like POS tags, word case, answer or its position as input, we generate our own named entity labels to be used as additional information in the input.

C. *Question Generation using Transformers*

Transformers [34] have been at the heart of some of the recent research in question generation. Chan and Fan (2019) [35] used BERT; Scialom et al. (2019) [36] used transformer on SQuAD for answer-agnostic question generation. Transformers have also been used for visual question generation by Matsumori et al. (2021) [37]. Our work uses the transformer called T5 (Text-To-Text Transfer Transformer) proposed by Raffel et al. (2020) [15].

III. METHODOLOGY

This section provides a formal definition of our task, followed by an overview of the deep learning models used in this work, and finally, the process of creating named entities for privacy policy documents.

A. Problem Definition

Question generation is defined as follows:

For a passage, $X_p = (x_1, x_2, ..., x_n)$ from a policy document, the model generates a question, Y = $(y_1, y_2, ..., y_T)$. The aim is to find the best $\bar{Y}$:

$$\bar{Y} = \underset{x}{\operatorname{argmax}} \; P(Y|X_p)$$

where $P(Y|X_p)$ is the conditional log-likelihood of the predicted question sequence y, given the input x. The answer or its position in the passage is not used as input.

The reason for not using the answer sequence as input lies within the nature of the data set, where many examples do not contain the answer verbatim or have answers that can only be inferred by a human on the basis of the question. The existing sequence-to-sequence models hard code answer positions or use answers, neither of which works well in this setting. So, we train all our models in an answer-unaware setting. In terms of novel contributions of this work, we generate our named entities for the privacy policy domain and then use them as supplementary input to the T5 and sequence-to-sequence models. This work compares the performance of sequence-to-sequence models to T5 transformer model, and assesses the performance benefits provided by our domain specific NER tags towards the results. It is part of an ongoing work, which aims to design a constraint-based question generation approach that uses T5 and sequence-to-sequence models and compares the two in terms of performance.

B. Deep Learning Models

Encoder + NER Labels - A uni-directional gated recurrent unit (GRU) [38]) forms the encoder. The encoder input is a concatenation of the two embedding vectors: word and label, just like the work of Zhou et al. (2017) [14], but unlike them the answer position or other lexical features like POS tags are not used. The output produced by an encoder serves as the decoder input.

Decoder - A uni-directional GRU decoder is used to generate the questions after decoding the label and context information from the encoder.

Decoder: Attention-Based - An attention-based GRU decoder [27] is used to generate the questions after decoding the label and context information from the encoder.

Transformer + NER Labels - Hugging Face library [39] is used in this work for T5 small model [15]. This model comprises of sixty million parameters. We use context information augmented with NER labels as input to the T5 model to generate questions.

C. Named Entities

All existing NER tools in the legal domain are specific to applications: court cases [40] [41], or German-language court cases [42]. We define 5 entity types with the aim of extracting important information from privacy policy documents. Table II presents these labels. We use the lookup method for named entity recognition due to lack of human annotators and training data. This method involves creating a list of entities and identifying all mentions of the list elements in the data set and marking them as entities. For example, every mention of the word "data" in the data set is marked as CONTENT. This method offers some advantages: simple implementation, easier maintenance, and it works in the absence of training data. A major disadvantage of this method is that it is plagued with false positives. Hence, we tried to make a very comprehensive list.

TABLE II
ENTITY LABELS

Label	Description
URL	Web link for the organization
PERSON	Words that are common nouns that refer to humans using the service, like customers, visitors, etc.
ORG	Words that are the names of organizations, and words like third party or vendors
CONTENT	Words that use the word "information" and the sources used for collecting it, for instance, words like cookies, browser, apps, and websites
LEGAL	Words that refer to policies or agreements

TABLE III
DISTRIBUTION OF DATA

Item Set	#Items
Training	20,013
Dev	2,502
Test	2,502

IV. Experiment Design

A. Data Set

This work uses the PolicyQA data set [9] which contains 25,017 reading comprehension style examples. This data set consists of privacy policies from 115 websites and includes 714 human-annotated questions. The data set is first shuffled and then split into three sets at the paragraph level: training, development and test set. The distribution of the data into the three sets is given in Table III.

B. Implementation Details

This work uses Nvidia Tesla v100 and PyTorch v1.7.1. for training all models. The following pre-processing steps have been performed:

- Data set is converted to lowercase.
- All questions are appended with SOS and EOS tokens.
- To maintain consistency throughout the data, all short forms were expanded and all observed spelling disparities were addressed. For instance, the word "info" was expanded to its full-form " information".

We experimented with various hidden sizes for the GRU of both the encoder and decoder: 500, 1000, and 2000. When using beam search for decoding, the beam width was set to 3.

TABLE IV
RESULTS (%): GREEDY SEARCH

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L
Learning to Ask (2017) [25]	**32.66**	**18.27**	**12.73**	**9.94**	15.54	30.63
Seq2Seq+attn	28.09	16.00	9.86	6.30	14.96	**31.84**
Vanilla Seq2Seq	27.22	15.52	8.63	5.12	14.23	30.16
T5-small model [15]	31.32	17.14	11.51	8.53	**18.29**	31.02

TABLE V
RESULTS (%): BEAM SEARCH

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L
Learning to Ask (2017) [25]	26.34	13.76	8.17	5.16	17.79	**35.25**
Seq2Seq+attn	**28.29**	**15.97**	**9.72**	6.36	16.15	30.02
Vanilla Seq2Seq	25.18	13.92	7.79	3.86	16.16	30.86
T5-small model [15]	28.19	14.77	9.51	**6.59**	**18.16**	28.85

TABLE VI
RESULTS (%) WITH OUR APPROACH VS TABLE IV

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L
Seq2Seq+attn	28.09	16.00	9.86	6.30	14.96	31.84
Vanilla Seq2Seq	27.22	15.52	8.63	5.12	14.23	30.16
T5 small model [15]	31.32	17.14	11.51	8.53	18.29	31.02
Seq2Seq+attn + NER	28.68	16.24	9.95	6.87	15.81	31.51
Vanilla Seq2Seq + NER	24.66	14.33	8.96	5.80	14.93	**33.07**
T5-small model + NER	**32.98**	**18.22**	**11.89**	**8.49**	**18.74**	32.16

SGD [43] was used for optimization with a 0.001 learning rate. We use teacher forcing for training and use the development set for hyper-parameter tuning and for selecting the best model based on the lowest model perplexity.

C. Baselines

The following baseline models have been used for comparison with our proposed approach:

- **Learning to Ask (2017)** [25]: global attention-based LSTM encoder-decoder model. We got the best result with paragraph level input and GloVe [44] embedding.
- **Seq2Seq + attention** [27]: encoder-decoder architecture with Bahdanau attention [27].
- **Vanilla Seq2Seq** [26]: basic encoder-decoder architecture. We do not reverse the input to this model.
- **Transformer-based model (T5)**: It is pre-trained on Colossal Clean Crawled Corpus [15]. We fine-tuned T5-small model for our task using a privacy policy passage as input.

D. Evaluation Metrics

All models in this work have been evaluated using package released by Chen et al. (2015) [45] for calculating ROUGE-L, METEOR, and BLEU-n scores. A high score indicates a better quality of the generated question.

V. RESULTS AND DISCUSSION

Tables IV and V present all our baseline models using greedy and beam search for decoding, respectively. Our goal was to see how performance (in terms of the evaluation metrics) varies among sequence-to-sequence and T5. We can clearly see that in both tables T5 small model gets a higher METEOR score of 18.29 using greedy search and 18.16 using beam search. The Rouge-L score for T5 model is quite lower as compared to sequence-to-sequence models. Attention-based sequence-to-sequence models give a better ROUGE-L and BLEU scores. Table VI presents our final results where the input to the models is a concatenation of the context with the named entity labels that have been presented in Section III of this paper. The results show that our provided input produces an improvement in each of the models, when compared with their corresponding baselines. The METEOR and ROUGE-L scores of the T5-small model with labels increase by 2.46 percent and 3.67 percent respectively. Vanilla Seq2Seq also shows an improvement of 4.9 percent in METEOR and 9.6 percent in ROUGE-L. Seq2Seq+attention also shows an improvement in METEOR with 5.6 percent increase over the baseline. However, it fails to beat the baseline score in terms of ROUGE-L. Additionally, Seq2Seq with attention shows 1 percentage point improvement over the vanilla Seq2Seq in terms of METEOR and BLEU, when named entity labels are provided to both. These results give us motivation to seek other ways of adding more background knowledge to these models in order to generate better results in terms of these metrics.

VI. Conclusion

This research presents the first milestone in our work of using privacy policy documents to automatically generate questions with the ultimate aim of building better QA systems. We compared the performance of sequence-to-sequence models with that of T5 model, which is a transformer model, using both decoding mechanisms of greedy and beam search. We generated our own named entity labels and used them as additional information to the models. Augmenting the context with labels gives the results a boost over the baseline models. In the future, we are interested in exploring more transformer-based architectures for this domain, and in enhancing the models by adding more background knowledge to produce diverse and syntactically and semantically better questions.

Acknowledgment

This work is part of the Ph.D. dissertation of Dr. Deepti Lamba [46].

References

[1] V. Rus, Z. Cai, and A. Graesser, "Question generation: Example of a multi-year evaluation campaign," *Proc WS on the QGSTEC*, 2008.

[2] M. Heilman and N. A. Smith, "Good question! statistical ranking for question generation," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010, pp. 609–617.

[3] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, "A systematic review of automatic question generation for educational purposes," *International Journal of Artificial Intelligence in Education*, vol. 30, no. 1, pp. 121–204, 2020.

[4] N. Mostafazadeh, I. Misra, J. Devlin, M. Mitchell, X. He, and L. Vanderwende, "Generating natural questions about an image," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1802–1813. [Online]. Available: https://www.aclweb.org/anthology/P16-1170

[5] N. Duan, D. Tang, P. Chen, and M. Zhou, "Question generation for question answering," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 866–874.

[6] D. Tang, N. Duan, T. Qin, Z. Yan, and M. Zhou, "Question answering and question generation as dual tasks," *arXiv preprint arXiv:1706.02027*, 2017.

[7] T. Wang, X. Yuan, and A. Trischler, "A joint model for question answering and question generation," *arXiv preprint arXiv:1706.01450*, 2017.

[8] A. Ravichander, A. W. Black, S. Wilson, T. Norton, and N. Sadeh, "Question answering for privacy policies: Combining computational and legal perspectives," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 4947–4958. [Online]. Available: https://www.aclweb.org/anthology/D19-1500

[9] W. U. Ahmad, J. Chi, Y. Tian, and K.-W. Chang, "Policyqa: A reading comprehension dataset for privacy policies," *arXiv preprint arXiv:2010.02557*, 2020.

[10] J. A. Obar and A. Oeldorf-Hirsch, "The biggest lie on the internet: Ignoring the privacy policies and terms of service policies of social networking services," *Information, Communication & Society*, vol. 23, no. 1, pp. 128–147, 2020.

[11] A. M. McDonald and L. F. Cranor, "The cost of reading privacy policies," *Isjlp*, vol. 4, p. 543, 2008.

[12] J. R. Reidenberg, T. Breaux, L. F. Cranor, B. French, A. Grannis, J. T. Graves, F. Liu, A. McDonald, T. B. Norton, and R. Ramanath, "Disagreeable privacy policies: Mismatches between meaning and users' understanding," *Berkeley Tech. LJ*, vol. 30, p. 39, 2015.

[13] F. H. Cate, "The failure of fair information practice principles," in *Consumer Protection in the Age of the'Information Economy'*. Routledge, 2016, pp. 351–388.

[14] Q. Zhou, N. Yang, F. Wei, C. Tan, H. Bao, and M. Zhou, "Neural question generation from text: A preliminary study," in *National CCF Conference on Natural Language Processing and Chinese Computing*. Springer, 2017, pp. 662–671.

[15] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html

[16] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[17] A. Lavie and M. J. Denkowski, "The meteor metric for automatic evaluation of machine translation," *Machine translation*, vol. 23, no. 2-3, pp. 105–115, 2009.

[18] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[19] K. Dhole and C. D. Manning, "Syn-QG: Syntactic and shallow semantic rules for question generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 752–765. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.69

[20] P. Khullar, K. Rachna, M. Hase, and M. Shrivastava, "Automatic question generation using relative pronouns and adverbs," in *Proceedings of ACL 2018, Student Research Workshop*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 153–158. [Online]. Available: https://www.aclweb.org/anthology/P18-3022

[21] Y. Chali and S. A. Hasan, "Towards Topic-to-Question Generation," *Computational Linguistics*, vol. 41, no. 1, pp. 1–20, 03 2015. [Online]. Available: https://doi.org/10.1162/COLI_a_00206

[22] M. Heilman and N. A. Smith, "Question generation via overgenerating transformations and ranking," Carnegie-Mellon Univ Pittsburgh pa language technologies insT, Tech. Rep., 2009.

[23] D. M. Gates, "Automatically generating reading comprehension look-back strategy: Questions from expository texts," CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, Tech. Rep., 2008.

[24] R. Mitkov, "Computer-aided generation of multiple-choice tests," in *International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003*, 2003, pp. 15–.

[25] X. Du, J. Shao, and C. Cardie, "Learning to ask: Neural question generation for reading comprehension," *arXiv preprint arXiv:1705.00106*, 2017.

[26] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *arXiv preprint arXiv:1409.3215*, 2014.

[27] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[28] L. Pan, W. Lei, T.-S. Chua, and M.-Y. Kan, "Recent advances in neural question generation," *arXiv preprint arXiv:1905.08949*, 2019.
[29] E. Brill, "A simple rule-based part of speech tagger," PENNSYLVANIA UNIV PHILADELPHIA DEPT OF COMPUTER AND INFORMATION SCIENCE, Tech. Rep., 1992.
[30] X. Ma, Q. Zhu, Y. Zhou, and X. Li, "Improving question generation with sentence-level semantic matching and answer position inferring," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 8464–8471.
[31] V. Harrison and M. Walker, "Neural generation of diverse questions using answer focus, contextual and linguistic features," *arXiv preprint arXiv:1809.02637*, 2018.
[32] Y. Kim, H. Lee, J. Shin, and K. Jung, "Improving neural question generation using answer separation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6602–6609.
[33] L. Song, Z. Wang, W. Hamza, Y. Zhang, and D. Gildea, "Leveraging context information for natural question generation," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018, pp. 569–574.
[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.
[35] Y.-H. Chan and Y.-C. Fan, "A recurrent BERT-based model for question generation," in *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 154–162. [Online]. Available: https://aclanthology.org/D19-5821
[36] T. Scialom, B. Piwowarski, and J. Staiano, "Self-attention architectures for answer-agnostic neural question generation," in *Proceedings of the 57th annual meeting of the Association for Computational Linguistics*, 2019, pp. 6027–6032.
[37] S. Matsumori, K. Shingyouchi, Y. Abe, Y. Fukuchi, K. Sugiura, and M. Imai, "Unified questioner transformer for descriptive question generation in goal-oriented visual dialogue," *arXiv preprint arXiv:2106.15550*, 2021.
[38] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734.
[39] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-demos.6
[40] S. Skylaki, A. Oskooei, O. Bari, N. Herger, and Z. Kriegman, "Named entity recognition in the legal domain using a pointer generator network," *arXiv preprint arXiv:2012.09936*, 2020.
[41] C. Dozier, R. Kondadadi, M. Light, A. Vachher, S. Veeramachaneni, and R. Wudali, "Named entity recognition and resolution in legal text," in *Semantic Processing of Legal Texts*. Springer, 2010, pp. 27–43.
[42] E. Leitner, G. Rehm, and J. Moreno-Schneider, "Fine-grained named entity recognition in legal documents," in *Semantic Systems. The Power of AI and Knowledge Graphs*, M. Acosta, P. Cudré-Mauroux, M. Maleshkova, T. Pellegrini, H. Sack, and Y. Sure-Vetter, Eds. Cham: Springer International Publishing, 2019, pp. 272–287.
[43] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.
[44] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: http://www.aclweb.org/anthology/D14-1162
[45] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick, "Microsoft coco captions: Data collection and evaluation server," *arXiv preprint arXiv:1504.00325*, 2015.
[46] D. Lamba, "Deep learning with constraints for answer-agnostic question generation in legal text understanding," Ph.D. dissertation, Kansas State University, Manhattan, Kansas, U.S.A., 2021. [Online]. Available: https://krex.k-state.edu/dspace/handle/2097/41629