

# A Multi-task Approach to Explaining Clarification Questions for Task-Oriented Conversational Systems

Jerome Ramos<sup>1</sup>

<sup>1</sup>University College London

## Abstract

Recent advances in artificial intelligence have increased interest in conversational systems in a variety of domains. As these conversational systems become more ubiquitous in everyday life, it becomes increasingly imperative that they are explainable in order to bridge the gap between human users and machine learning models. In this work, we investigate the trade-offs between non-explainable and explainable conversational systems. In particular, we train an explainable model to complete the ShARC conversational machine reading task and compare its performance to non-explainable, state-of-the-art models. In addition, we discuss future avenues of work in the domain of explainable conversational systems.

## Keywords

Conversational Systems, Transparency and Explainability, Recommender Systems

## 1. Introduction

In recent years, researchers have focused on improving chatbots to handle complex tasks in a variety of domains such as e-commerce [1], music recommendation [2], and general chit-chat [3]. Due to the growing complexity of these models, many researchers have begun to focus on the topic of explainability and transparency. Research has shown that explainability improves trustworthiness, effectiveness, efficiency, and user satisfaction [4]. Providing transparency for why a chatbot returned a particular answer can help users understand the system's reasoning and potentially scrutinize and correct the model. We present our current work on explainable conversational systems through extractive explanations in conversational machine reasoning, a sub-type of task-oriented dialogue systems. In particular, we show that an explainable conversational machine reading system performs comparably to its non-explainable counterparts. We also present our roadmap for future work including:

1. Training multi-task learning for explainable conversational recommender systems using ideas learned from the explainable conversational machine reading task.
2. Developing generative explanations for conversational systems.
3. Creating a framework to enable scrutability and allow users to provide feedback to conversational systems.
4. Creating novel metrics to measure the efficacy of the generated explanations.


---

FDIA 2022, 20 July 2022, Lisbon, Portugal.

✉ [jerome.ramos.20@ucl.ac.uk](mailto:jerome.ramos.20@ucl.ac.uk) (J. Ramos)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings ([CEUR-WS.org](http://CEUR-WS.org))

## 2. Related Works

### 2.1. Task-oriented Dialogue Systems

In recent years, task-oriented dialogue systems have received considerable interest [5]. Unlike open-domain dialogue systems, which do not have any predefined tasks, task-oriented dialogue systems are developed for specific domains [6]. There are several types of research problems in dialogue systems. Question Answering (QA) tries to answer a question given a passage with two different settings: *Extractive QA*, where the answer is inside the passage; and *Generative QA*, where the answer must be generated on the fly [7].

Similarly, Conversational Machine Reading (CMR) requires the agent to extract information from the passage and be able to answer questions in a conversational manner [8]. Previous works on the ShARC CMR dataset [9] use various techniques such as dialogue graph modeling [10], discourse-aware entailment reasoning [11], and explicit memory tracking [12]. Although these models can answer a user’s question with high accuracy, they are unable to explain how it arrived at that answer. This paper aims to borrow techniques from existing task-oriented dialogue systems and combine them with an extractive explanation module to provide both a performative and transparent conversational system.

### 2.2. Transparency and Explainability

Due to the increasing complexity of machine learning models, many researchers have become interested in developing models that are transparent and explainable. In the context of dialogue systems, an explainable conversational agent would not only be able to respond to a user’s utterance, but also inform the user why it responded the way it did. In addition to increasing user confidence (trustworthiness), explainable AI can help users make decisions better (effectiveness) and faster (efficiency), as well as increase the ease of use of a system (satisfaction) [4].

In general, there are two types of explainable machine learning models: post-hoc explanations and inherently interpretable models. Post-hoc explanations provide explanations post-training. Counterfactual explanations are trained to show how changing one or more than one feature value in a data point will lead to a different prediction in the model [13]. Model agnostic methods such as LIME [14] and SHAP [15] can generate explanations for any black-box machine learning algorithm. In contrast, there have been several approaches to inherently interpretable models such as set-based techniques [16] and additive models [17, 18]. In this work, we chose the direction of inherently interpretable models. To our knowledge, this is the first work in task-oriented dialogue systems that provides explainability via multi-span rule text extraction.

## 3. Explainable Conversational Systems

Conversational Machine Reading (CMR) systems are a type of task-oriented dialogue systems that are designed to answer a user’s high-level questions by interpreting natural language rules and asking the user clarification questions. For example, in Table 1, the user asks, “Can I get Public Housing?”. In order to determine if the user is eligible for public housing, the conversational agent must extract rules from the rule text. The system must then learn which

---

|                   |  |
|-------------------|--|
| <b>Rule Text:</b> | Public Housing provides affordable rental houses or apartments for low-income families, people who are elderly, and people with disabilities. To apply, contact a public housing agency in your state. |
|-------------------|--|

---

|                                    |   |
|------------------------------------|---|
| <b>User Scenario:</b>              | I am a 45 year old man who does not want to decrease it up to 13 weeks under any circumstances. |
| <b>Question:</b>                   | Can I get Public Housing?   |
| <b>Follow-up <math>Q_1</math>:</b> | Are you a low-income family?  |
| <b>Follow-up <math>A_1</math>:</b> | No  |
| <b>Follow-up <math>Q_2</math>:</b> | Are you elderly?  |
| <b>Follow-up <math>A_2</math>:</b> | No  |

---

|                      |  |
|----------------------|--|
| <b>Model Output:</b> | Do you have a disability?  |
| <b>Explanation:</b>  | Public Housing provides affordable rental houses or apartments for people with disabilities. |

---

**Table 1**

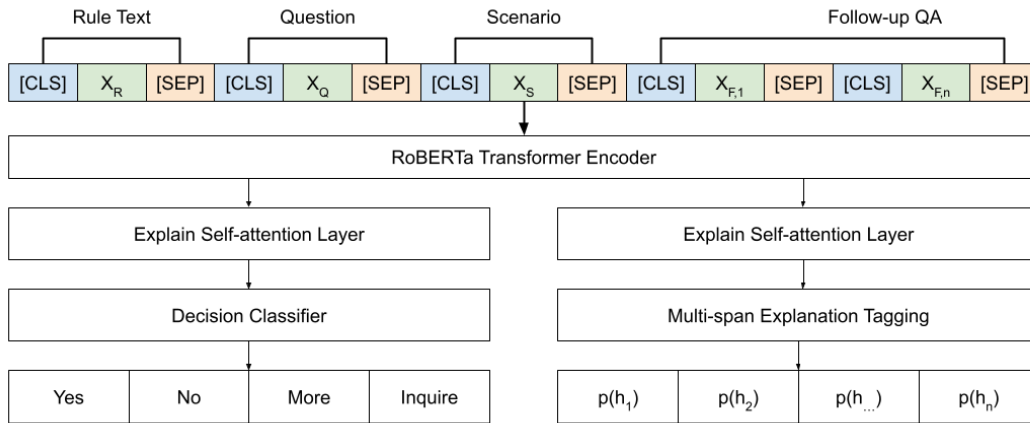
Example conversation in the ShARC dataset. Given a rule text, user scenario, and dialogue history, the system must determine if the user’s question can be answered or if an additional follow-up question must be generated. The model is also trained to highlight the words in the rule text that can explain the model’s output. In the case of Inquire, as is shown in the example above, the system generates a follow-up question and a corresponding explanation extracted from the rule text, which is highlighted in blue.

of the extracted rules have already been covered in the users scenario, a synopsis of the user’s circumstances which may or may not implicitly contain rules, and the dialogue history, where the system explicitly asks the user if they meet the condition of a rule in the form of a “Yes” or “No” question. Finally, the system must determine if it can answer the user’s question or if an additional clarification question is needed.

For our preliminary work, we worked on solving the task outlined in the ShARC dataset [9]. The current state-of-the-art approaches [10, 11, 12] split the problem into two tasks. The first task is to classify the decision as “Yes”, “No”, “Irrelevant”, or “Inquire”. The “Yes” and “No” responses directly answer the user’s question, whereas “Irrelevant” means that the user’s question cannot be answered by the rule text. If the system classifies the conversation as “Inquire”, the system must gather additional information from the user before returning a final answer. This is generally done by extracting a span from the rule text and transforming the under-specified span into a question.

Although current approaches are capable of highlighting underspecified spans in the rule text, they are not explainable to users because the span is not a complete sentence. For example, state-of-the-art approaches would highlight “people with disabilities” in Table 1 as an underspecified span to transform into a question. This can help developers debug the model, but would not be a good explanation for end users because the underspecified span does not make sense on its own. Thus, it is imperative to extract a complete sentence from the rule text in order for the model to be considered explainable and transparent to users. Providing explainability can improve user trust and allow users to better understand how the system generates its answers. Additionally, explanations can also help users understand the relevance of the clarification question and help guide them through the conversation.

In order to have a ground-truth dataset for extracted explanations, we developed a crowd-



**Figure 1:** Decision Making and Explanation Model

sourcing web application to annotate extractive explanations for each clarification question. For each question, crowd workers were tasked with highlighting the portions of the rule text that explains why the clarification is relevant to the conversation’s high-level question. Crowd workers were encouraged to highlight complete sentences where possible so that the explanation could be read in a natural way. The new dataset consists of question-explanation pairs for every follow-up question in the ShARC dataset.

Once the annotation was complete, we trained a new baseline model to classify decisions, generate follow up questions if necessary, and extract explanations from the rule text. We show that the explainable model performs comparably to non-explainable, state-of-the-art models. In summary, we provide a new corpus of explanations for over 32k conversations in the ShARC dataset and we developed a novel model for that can complete the original ShARC task and simultaneously extract explanations from the rule text.

## 4. Model

Our system consists of two separate models that are trained for two different tasks. We first tokenize the user question, user scenario, user history, and dialogue history as inputs. We then train a multi-task model for decision classification and explanation extraction. For decisions that are classified as “Inquire”, we train a second model to transform a chosen underspecified span in the rule text into a follow-up question.

### 4.1. Decision Module

The decision module is responsible for determining the type of answer for the user’s question. The system can respond with 4 classes: Yes, No, Irrelevant, Inquire. Let  $x_R$ ,  $x_Q$ ,  $x_S$ ,  $x_{F,i}$ , be the rule text, user question, user scenario, and the follow-up question and answer on the  $i$ th

turn. We concatenate these inputs into a single sequence  $[x_R, x_Q, x_S, x_{F,i}]$ . We then add a [CLS] token at the start of each type of input and a [SEP] token at the end of each type of input. [CLS] and [SEP] tokens are also added in the same manner between every  $i$ th turn of  $x_{F,i}$ . We then encode the concatenated sequence using RoBERTa-base with the default dimension of  $d = 768$  and compute a summary  $C$  using self-attention

$$\phi_k = \text{softmax}(W_\phi U_k + b_\phi)_k \in \mathbb{R} \quad (1)$$

$$C = \sum_{k=s_i}^{e^i} \phi_k U_k \in \mathbb{R}^{d_U} \quad (2)$$

where  $W_\phi \in \mathbb{R}^{d_U}$ ,  $b_\phi \in \mathbb{R}$  and  $\phi$  is the normalized self-attention weights. We then calculate the scores for each choice and select the one of the decision classes, Yes, No, Irrelevant, and Inquire, by computing

$$s = W_s C + b_s \in \mathbb{R}^4 \quad (3)$$

$$d = \text{argmax}_k s_k \in \mathbb{R}^4 \quad (4)$$

where  $s$  is the score vector for each of the four decisions classes and  $d$  is the selected decision.

## 4.2. Explanation Module

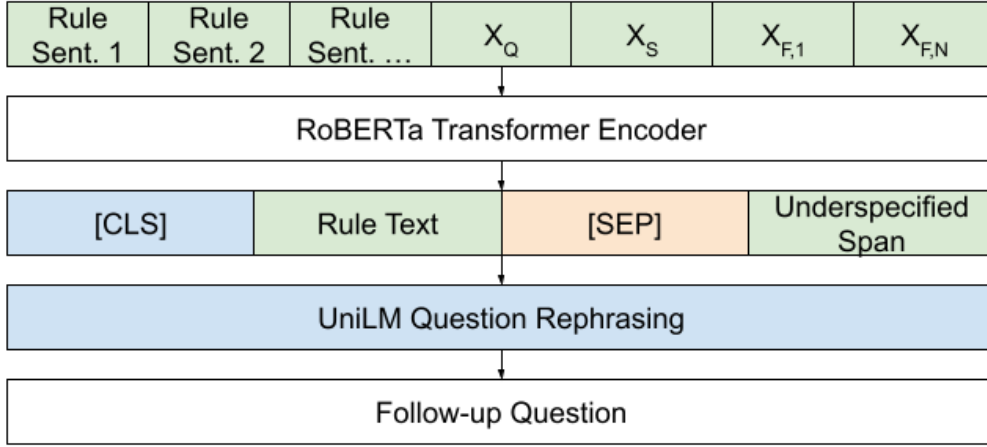
Although a single-span model used in previous works [11, 12, 19] performs well on rule span extraction, it would not perform well on explanation extraction because many explanations are multi-span. Thus, we use a multi-span extraction method as described in [20]. Similar to the sequence tagging problem, a common task in Named Entity Recognition (NER), the multi-span model outputs a probability distribution over a set of tags for each token. We experiment using IO-tagging to determine whether a token is inside (I) the answer or outside (O). Let  $h$  be the same contextualized representation as mentioned in the single-span model. For every token, we calculate the probability of the tag of the  $i$ -th token as

$$p_i = \text{softmax}(f(h_i)) \quad (5)$$

where  $p \in \mathbb{R}^{m \times |S|}$  and  $f$  is a parameterized function with  $|S|$  outputs. Training is done by minimizing the binary cross entropy. During test time, the predicted tag for each token is selected by choosing the tag with the highest probability.

We implement the multi-span model as a second task for the model to train for, along with the aforementioned decision classification module. If the decision is ‘‘Inquire’’, then the objective of the model is to extract the explanation for the follow-up question that should be generated. If the decision is ‘‘Yes’’ or ‘‘No’’, then the objective is to extract the explanation for why the system came to that conclusion. The explanation for irrelevant decisions is an empty string as there is no explanation for an irrelevant case. After generating the explanation for each conversation, each token that is part of the explanation is labelled as (I) and all other tokens that are not part of the explanation are labelled as (O). The model is then trained to correctly tag each token as either (I) or (O). The total loss  $\mathcal{L}$  of the model is calculated as:

$$\mathcal{L} = \mathcal{L}_{\text{decision}} + \mathcal{L}_{\text{explanation}} \quad (6)$$



**Figure 2:** Question Generation Model

### 4.3. Question Generation Module

We follow the question generation techniques as described in [11]. All decisions classified as Inquire are passed to the question generation model as shown in Figure 2. The rule text is first split up into rule sentences. The resulting sentences are then concatenated with the question, user scenario, and the user’s dialogue history. The sequence is then encoded using RoBERTa-base and transformed into tokens. The model is trained to learn a start vector  $w_s \in \mathbb{R}^d$  and end vector  $w_e \in \mathbb{R}^d$  with the constraint that the start and end positions are in the same rule sentence:

$$Span = \underset{i,j,k}{argmax} (w_s^\top t_{k,i} + w_e^\top t_{k,j}) \quad (7)$$

where  $i, j$  are the start and end positions of the chosen span and  $k$  is the sentence which contains the span. The model is trained to maximize the sum of the log-likelihood of the correct start and end positions. Since there is no ground-truth span dataset, a span supervision dataset is noisily generated by selecting the minimum edit distance of the each follow-up question in order to extract spans from the rule text. Finally, we concatenate the rule text and chosen underspecified span and finetune UniLM [7] to rephrase the span into a question.

## 5. Results

The performance of the decision classification and question generation sub-tasks can be seen in Table 2. The results show that the micro-accuracy and macro-accuracy of the explainable model is similar to that of state-of-the-art, non-explainable models. Given the simplicity of the

| Models       | Micro Acc. | Macro Acc. | BLEU1 | BLEU4 |
|--------------|------------|------------|-------|-------|
| BERTQA [19]  | 68.6       | 73.7       | 47.4  | 54.0  |
| E3 [19]      | 68.0       | 73.4       | 66.9  | 53.7  |
| EMT [12]     | 73.2       | 78.3       | 67.5  | 53.2  |
| Discern [11] | 74.9       | 79.8       | 65.7  | 52.4  |
| DGM [10]     | 78.6       | 82.2       | 71.8  | 60.2  |
| Our Model    | 71.6       | 77.4       | 62.5  | 48.8  |

**Table 2**

Performance on the development set of the ShARC end-to-end task for the decision classification and question generation sub-tasks.

decision-making module of our model, it would be interesting to see if combining a state-of-the-art decision classifier with our explanation module would yield better performance. It would also be interesting to experiment with using the extracted explanation as a span to pass to the UniLM model rather than having a separate model trained to extract under-specified spans. Given that the explainable model slightly under-performs non-explainable models, developers of conversational systems would have to weigh the trade-offs between decision classification accuracy and transparency.

Additionally, we calculate the exact match (EM) and F1 scores for the extracted explanations of the dialogue history in Table 3. Note that for Irrelevant, the F1 score is omitted because there are no tokens to be highlighted, which always leads to an F1 score of 0. We also ran an additional annotation phase to check if the extracted explanations were of acceptable quality, which can be seen in the *Valid* column of Table 3. Annotators are shown the entire dialogue and rule text, as well as the model’s generated explanation. They are then asked whether they believe the explanation sufficiently answers why the model’s output is relevant to the user’s question.

The model outputs the best explanations the best when the decision class is “Yes” and the worst when the decision class is “No” across all three metrics. This suggests that the model can better explain why a user has met condition(s) better than why a user did not meet condition(s). Interestingly, the performance of “Inquire” lies between “Yes” and “No”. This means that the model understands which rule should be asked next and can generate the corresponding explanation. One major limitation of the explanation metrics is that EM and F1 captures whether the predicted explanation is similar to the labelled ground truth and does not measure whether the explanation is syntactically correct or is a sufficient explanation for human users. Additionally, manually verifying if explanations are sufficient is not a scalable metric. Thus, a scalable, algorithmic approach is needed to measure the correctness of the explanations.

## 6. Conclusion

In this paper, we present an explainable dialogue system for conversational machine reading. Our model runs multi-task learning to train for both decision classification and extractive explanation generation. We show that an explainable model has comparable performance to non-explainable, state-of-the-art models. The explanations returned can help users understand

| Class      | EM   | F1   | User Score |
|------------|------|------|------------|
| Yes        | 45.2 | 88.7 | 77.1       |
| No         | 33.8 | 82.5 | 65.8       |
| Inquire    | 37.4 | 84.0 | 70.0       |
| Irrelevant | 97.8 | -    | 97.8       |

**Table 3**

Extractive explanations for conversation. F1 score is omitted for Irrelevant because F1 score will always be zero as no tag should be labelled as (I).

the model’s response. In future work, we plan to use generative methods to provide natural language explanations to users. Additionally, we plan to explore how we can use multi-task learning as described in this paper and apply it to a conversational recommender system. We will also investigate how to make the model scrutable so that users can fix incorrect assumptions about their user scenario. Finally, we plan to design better metrics to measure the correctness of generated explanations.

## Acknowledgments

I would like to thank my advisor, Dr. Aldo Lipani, for his guidance and support on my research projects and Hossein A. Rahmani for his feedback on the paper.

## References

- [1] L. Cui, S. Huang, F. Wei, C. Tan, C. Duan, M. Zhou, SuperAgent: A customer service chatbot for E-commerce websites, in: Proceedings of ACL 2017, System Demonstrations, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 97–102. URL: <https://aclanthology.org/P17-4017>.
- [2] L. Zhou, J. Gao, D. Li, H.-Y. Shum, The design and implementation of XiaoIce, an empathetic social chatbot, Computational Linguistics 46 (2020) 53–93. URL: <https://aclanthology.org/2020.cl-1.2>. doi:10.1162/coli\_a\_00368.
- [3] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, S. Niu, DailyDialog: A manually labelled multi-turn dialogue dataset, in: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Asian Federation of Natural Language Processing, Taipei, Taiwan, 2017, pp. 986–995. URL: <https://aclanthology.org/I17-1099>.
- [4] N. Tintarev, J. Masthoff, Explaining recommendations: Design and evaluation, in: Recommender Systems Handbook, 2015.
- [5] Y. Feng, A. Lipani, F. Ye, Q. Zhang, E. Yilmaz, Dynamic schema graph fusion network for multi-domain dialogue state tracking, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 115–126. URL: <https://aclanthology.org/2022.acl-long.10>. doi:10.18653/v1/2022.acl-long.10.
- [6] M. Huang, X. Zhu, J. Gao, Challenges in building intelligent open-domain dialog systems, ACM Transactions on Information Systems (TOIS) 38 (2020) 1 – 32.



- [7] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, H.-W. Hon, Unified language model pre-training for natural language understanding and generation, *Advances in Neural Information Processing Systems* 32 (2019).
- [8] S. Gupta, B. P. S. Rawat, H. Yu, Conversational machine comprehension: a literature review, in: *Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020*, pp. 2739–2753. URL: <https://aclanthology.org/2020.coling-main.247>. doi:10.18653/v1/2020.coling-main.247.
- [9] M. Saeidi, M. Bartolo, P. Lewis, S. Singh, T. Rocktäschel, M. Sheldon, G. Bouchard, S. Riedel, Interpretation of natural language rules in conversational machine reading, *arXiv preprint arXiv:1809.01494* (2018).
- [10] S. Ouyang, Z. Zhang, H. Zhao, Dialogue graph modeling for conversational machine reading, in: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Association for Computational Linguistics, Online, 2021, pp. 3158–3169. URL: <https://aclanthology.org/2021.findings-acl.279>. doi:10.18653/v1/2021.findings-acl.279.
- [11] Y. Gao, C.-S. Wu, J. Li, S. Joty, S. C. Hoi, C. Xiong, I. King, M. Lyu, Discern: Discourse-aware entailment reasoning network for conversational machine reading, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 2439–2449. URL: <https://aclanthology.org/2020.emnlp-main.191>. doi:10.18653/v1/2020.emnlp-main.191.
- [12] Y. Gao, C.-S. Wu, S. Joty, C. Xiong, R. Socher, I. King, M. Lyu, S. C. Hoi, Explicit memory tracker with coarse-to-fine reasoning for conversational machine reading, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 935–945. URL: <https://aclanthology.org/2020.acl-main.88>. doi:10.18653/v1/2020.acl-main.88.
- [13] S. Verma, J. Dickerson, K. Hines, Counterfactual explanations for machine learning: A review, 2020. *arXiv:arXiv:2010.10596*.
- [14] M. Ribeiro, S. Singh, C. Guestrin, “why should I trust you?”: Explaining the predictions of any classifier, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, Association for Computational Linguistics, San Diego, California, 2016, pp. 97–101. URL: <https://aclanthology.org/N16-3020>. doi:10.18653/v1/N16-3020.
- [15] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 30, Curran Associates, Inc., 2017. URL: <https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf>.
- [16] K. Balog, F. Radlinski, S. Arakelyan, Transparent, scrutable and explainable user models for personalized recommendation, in: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR’19*, Association for Computing Machinery, New York, NY, USA, 2019, p. 265–274. URL: <https://doi.org/10.1145/3331184.3331211>. doi:10.1145/3331184.3331211.
- [17] Y. Lou, R. Caruana, J. Gehrke, G. Hooker, Accurate intelligible models with pairwise interactions, *Proceedings of the 19th ACM SIGKDD international conference on Knowledge*

- discovery and data mining (2013).
- [18] B. Ustun, S. Tracà, C. Rudin, Supersparse linear integer models for predictive scoring systems, in: AAAI, 2013.
  - [19] V. Zhong, L. Zettlemoyer, E3: Entailment-driven extracting and editing for conversational machine reading, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 2310–2320. URL: <https://aclanthology.org/P19-1223>. doi:10.18653/v1/P19-1223.
  - [20] E. Segal, A. Efrat, M. Shoham, A. Globerson, J. Berant, A simple and effective model for answering multi-span questions, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 3074–3080. URL: <https://aclanthology.org/2020.emnlp-main.248>. doi:10.18653/v1/2020.emnlp-main.248.