Supplementary Materials for EMNLP 2018 Paper:
Interpretation of Natural Language Rules in Conversational Machine Reading

A Annotation Interfaces

Figure 4 shows the Mechanical-Turk interface we developed for the dialog generation stage. Note that the interface also contains a mechanism to validate previous utterances in case they have been generated by different annotators.

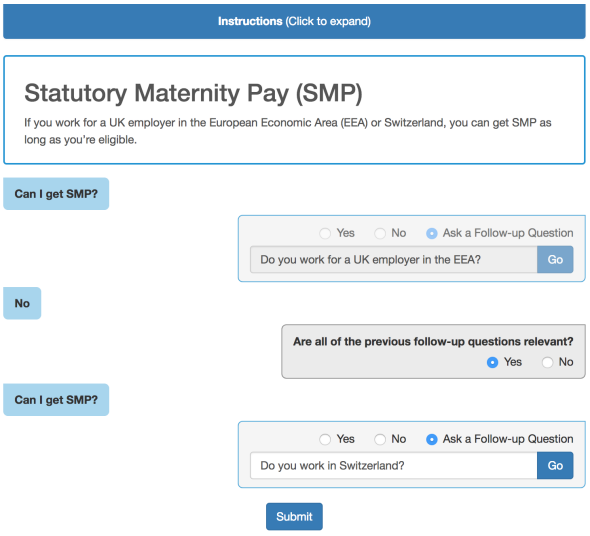


Figure 4: The dialog-style web interface encourages workers to extract all the rule text-relevant evidence required to answer the initial question in the form of YES/NO follow-up questions.

Figure 5 shows the annotation interface for the scenario generation task, where the first question is relevant and the second question is not relevant.

B Quality Control

In this section, we present several measure that we take in order to create a high quality dataset.

Irregularity Detection A convenient property of the formulation of the reasoning process as a binary decision tree is class exclusivity at the final partitioning of the utterance space. That is, if the two leaf nodes stemming from the same FOLLOW-UP QUESTION node have identical YES or NO values, this is an indication of either a mis-annotation or a redundant question. We automatically identify these irregularities, trim the subtree at FOLLOW-UP QUESTION node and re-annotate. This also means that our protocol effectively guarantees a minimum of two annotations per leaf node, further enhancing data quality.

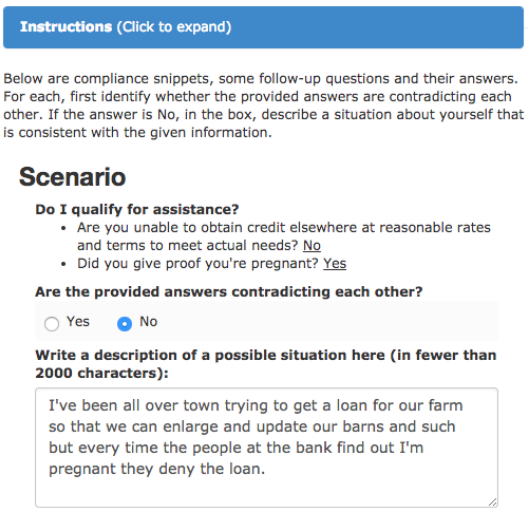


Figure 5: Annotators are asked to write a scenario that fits the given information, i.e. questions and answers.

Back-validation We implement back-validation by providing the workers with two options: YES and proceed with the task, or NO and provide an invalidation reason to de-incentivize unnecessary rejections. We found this approach to be valuable both as a validation mechanism as well as a means of collecting direct feedback about the task and the types of incorrect annotations encountered. We then trim any invalidated subtrees and re-annotate.

Contradiction Detection We can introduce contradictory information by adding random questions and answers to a dialog part when generating HITs for scenario generation. Therefore, we first ask each annotator to identify whether the provided dialog parts are contradictory. If they are, the annotator will invalidate the HIT.

Validation Sampling We sample a proportion of each worker's annotations to validate. Through this process, each worker is assigned a quality score. We only allow workers with a score higher than a certain value to participate in our HITs (Snow et al., 2008). We also restrict participation to workers with $> 97\%$ approval rate, > 1000 previously completed HITs and located in the UK, US or Canada.

Qualification Test Amazon Mechanical Turk allows the creation of qualification tests through the API, which need to be passed by each turker before

attempting any HIT from a specific task. A qualification can contain several questions with each having a value. The qualification requirement for a HIT can specify that the total value must be over a specific threshold for the turker to obtain that qualification. We set this threshold to 100%.

Possible Sources of Noise Here we detail possible sources of noise, estimate their effects and outline the steps taken to mitigate these sources:

a) Noise arising from annotation errors: This has been discussed in detail above.

b) Noise arising from negative question generation: Some noise could be introduced due to the automatic sampling of the negative questions. To obtain an estimate, 100 negative questions were assessed by an expert annotator. It was found that only 8% of negatively sampled questions were erroneous.

c) Noise arising from the negative scenario sampling: A further 100 utterances with negatively sampled scenarios were curated by an expert annotator, and it was found that 5% of the utterances were erroneous.

d) Errors arising from the application of scenarios to dialog trees: The assumption that the scenario was only relevant to the follow-up questions it was generated from, and was independent to all other follow-up questions posed in that dialog tree is not necessarily true, and could result in noisy dialog utterances. 100 utterances from the subset of the data where this type of error was possible were assessed by expert annotators, and 12% of these utterances were found to be erroneous. This type of error can only affect 80% of utterances, thus the estimated total effect of this type of noise is 10%.

Despite the relatively low levels of noise, we asked expert annotators to manually inspect and curate (if necessary) all the instances in the development and the test set that are prone to potential errors. This leads to an even higher quality of data in our dataset.

C Further Details on Corpus

We use 264 unique sources from 10 unique domains listed below. For transparency and reproducibility, the source URLs are included in the corpus for each dialog utterance.

- http://legislature.maine.gov/
- https://esd.wa.gov/
- https://www.benefits.gov/
- https://www.dmv.org/
- https://www.doh.wa.gov/
- https://www.gov.uk/
- https://www.humanservices.gov.au/
- https://www.irs.gov/
- https://www.usa.gov/
- https://www.uscis.gov/

Further, the ShARC dataset composition can be seen in Table 6.

Set	# Utterances	# Trees	# Scenarios	# Sources
All	32436	948	6637	264
Train	21890	628	4611	181
Development	2270	69	547	24
Test	8276	251	1910	59

Table 6: Dataset composition.

D Negative Data

In this section, we provide further details regarding the generation of the negative examples.

D.1 Negative Questions

Formally, for each unique positive question, rule text pair, (q_i, r_i), and defining d_i as the source document for (q_i, r_i), we construct the set $Q \in \{q_1 \dots q_n\}$ where Q is the set of questions that are not sourced from d_i. We take a random uniform sample q_j from Q to generate the negative utterance (q_j, r_i, h_j, y_j) where y_j = IRRELEVANT and h_j is an empty history sequence. An example of a negative question is shown below.

Q. Can I get Working Tax Credit?

R. You must also wear protective headgear if you are using a learner's permit or are within 1 year of obtaining a motorcycle license.

D.2 Negative Scenarios

We also negatively sample scenarios so that models can learn to ignore distracting scenario information that is not relevant to the task. We define a negative scenario as a scenario that provides no information to assist answering a given question and as such,

good models should ignore all details within these scenarios.

A scenario s_x is associated with the (one or more) dialog question and answer pairs $\{(f_{x,1}, a_{x,1}) .. (f_{x,n}, a_{x,n})\}$ that it was generated from.

For a given unique question, rule text pair, (q_i, r_i), associated with a set of positive scenarios $\{s_{i,1} \ldots s_{i,k}\}$, we uniformly randomly sample a candidate negative scenario s_j from the set of all possible scenarios. We then build TF-IDF representations for the set of all dialog questions associated with (q_i, r_i), i.e. $F_i = \{(f_{i,1,1}) .. (f_{i,k,n})\}$. We also construct TF-IDF representations for the set of dialog questions associated with s_j, $F_{s_j} = \{(f_{j,1}) .. (f_{j,x})\}$.

If the cosine similarity for all pairs of dialog questions between F_i and F_{s_j} are less than a certain threshold, the candidate is accepted as a negative, otherwise a new candidate is sampled and the process is repeated. Then we iterate over all utterances that contain (q_i, r_i) and use the negative scenario to create one more utterance whenever the original utterance has an empty scenario. The threshold value was validated using manual verification. An example is shown below:

R. You are allowed to make emergency calls to 911, and bluetooth devices can still be used while driving.

S. The person I'm referring to can no longer take care of their own affairs.

E Challenges

In this section we present a few interesting examples we encountered in order to provide a better understanding of the requirements and challenges of the proposed task.

E.1 Dialog Generation

Table 8 shows the breakdown of the types of challenges that exist in our dataset for dialog generation and their proportion.

F Entailment Corpus

Using the scenarios and their associated questions and answers we create an entailment corpus for each of the train, development and test sets of ShARC. For every dialog utterance that includes a scenario, we create a number of data points as follows:

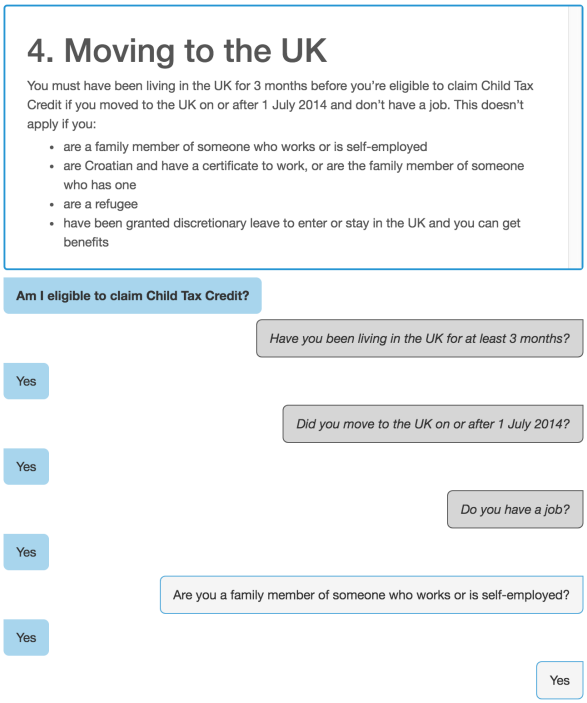


Figure 6: Example of a complex and hard-to-interpret rule relationship.

For every utterance in ShARC with input $x = (q, r, h, s)$ and output y where $y = f_m \notin \{\text{YES}, \text{NO}, \text{IRRELEVANT}\}$, we create an entailment instance (x_e, y_e) such that $x_e = s$ and:

- y_e = ENTAILMENT if the answer a_m to follow-up question f_m is YES which can be derived from s.
- y_e = CONTRADICTION if the answer a_m to follow-up question f_m is NO which can be derived from s.
- y_e = NEUTRAL if the answer a_m to follow-up question f_m cannot be derived from s.

Table 7 shows the statistics for the entailment corpus.

Set	ENTAILMENT	CONTRADICTION	NEUTRAL
Train	2373	2296	10912
Dev	271	253	1098
Test	919	944	4003

Table 7: Statistics of the entailment corpus created from the ShARC dataset.

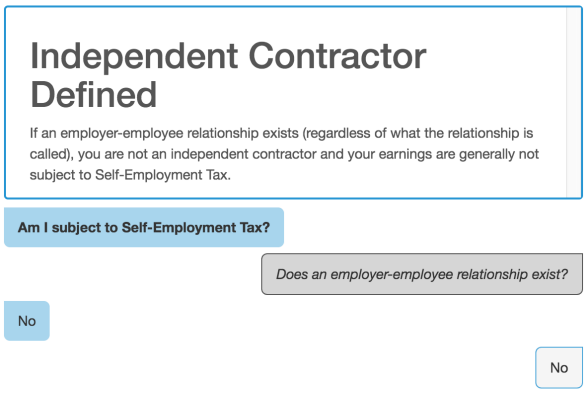


Figure 7: Example of a hard-to-interpret rule due to complex negations. In this particular example, majority vote was inaccurate.

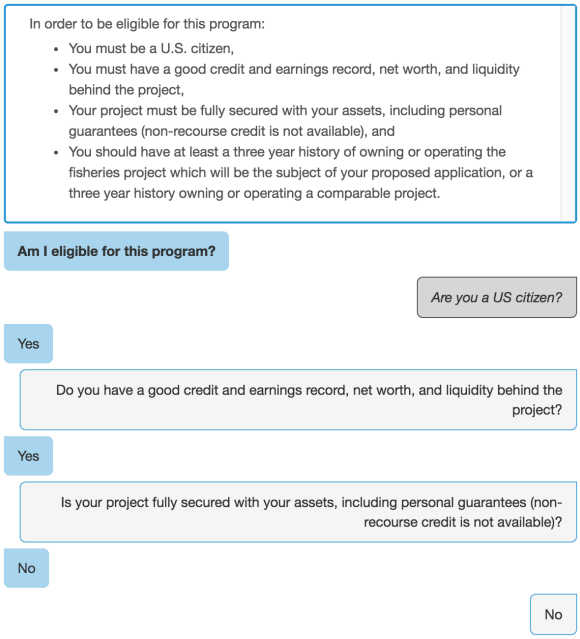


Figure 8: Example of a conjunctive rule relationship derived from a bulleted list, determined by the presence of “, and” in the third bullet.

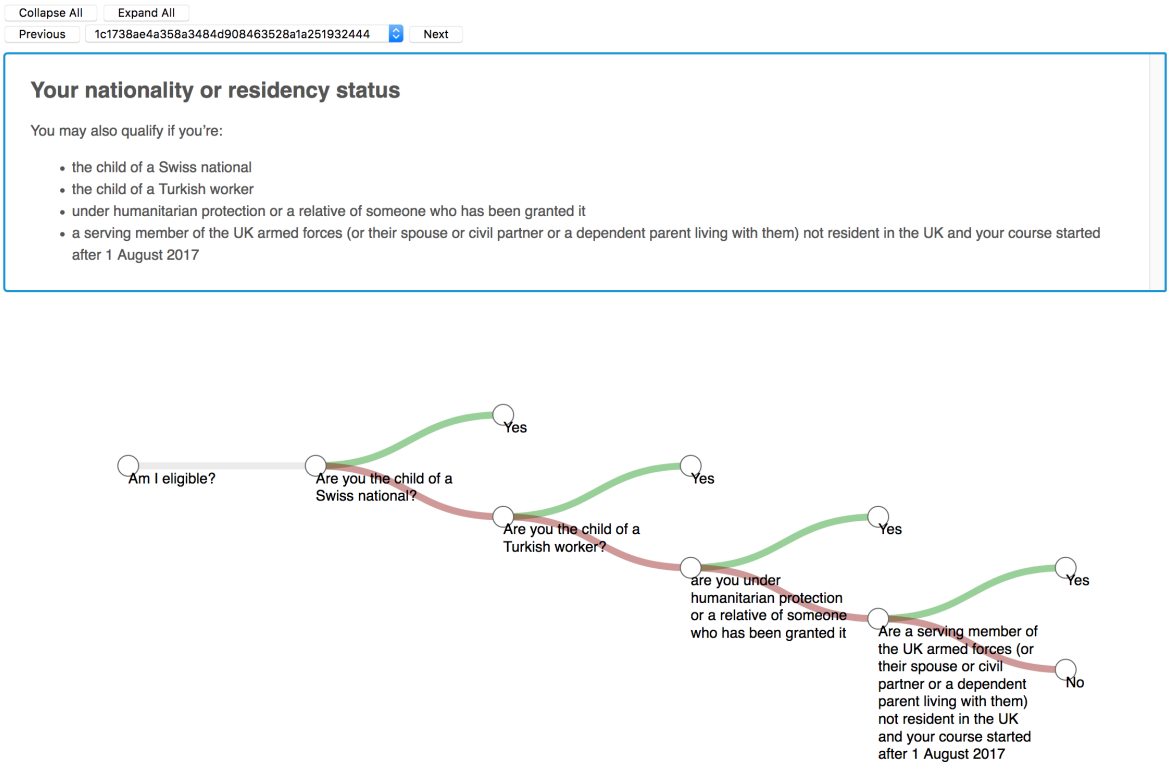


Figure 9: Example of a dialog-tree for a typical disjunctive bulleted list.

G Further details on Interpreting rules

Category	Example Question	Example Rule Text	Percentage
Simple	Can I claim extra MBS items?	If youre providing a bulk billed service to a patient you may claim extra MBS items.	31%
Bullet Points	Do I qualify for assistance?	To qualify for assistance, applicants must meet all loan eligibility requirements including: • Be unable to obtain credit elsewhere at reasonable rates and terms to meet actual needs; • Possess legal capacity to incur loan obligations;	34%
In-line Conditions	Do these benefits apply to me?	These are benefits that apply to individuals who have earned enough Social Security credits and are at least age 62.	39%
Conjunctions	Could I qualify for Letting Relief?	If you qualify for Private Residence Relief and have a chargeable gain, you may also qualify for Letting Relief. This means youll pay less or no tax.	18%
Disjunctions	Can I get deported?	The United States may deport foreign nationals who participate in criminal acts, are a threat to public safety, or violate their visa.	41%
Understanding Questioner Role	Am I eligible?	The borrower must qualify for the portion of the loan used to purchase or refinance a home. Borrowers are not required to qualify on the portion of the loan used for making energy-efficient upgrades.	10%
Negations	Will I get the National Minimum Wage?	You wont get the National Minimum Wage or National Living Wage if youre work shadowing	15%
Conjunction Disjunction Combination	Can my partner and I claim working tax credit?	You can claim if you work less than 24 hours a week between you and one of the following applies: • you work at least 16 hours a week and youre disabled or aged 60 or above • you work at least 16 hours a week and your partner is incapacitated	18%
World Knowledge Required to Resolve Ambiguity	Do I qualify for Statutory Maternity Leave?	You qualify for Statutory Maternity Leave if: • youre an employee not a 'worker' • you give your employer the correct notice	13%

Table 8: Types of features present for question, rule text pairs and their proportions in the dataset based on 100 samples. World Knowledge Required to resolve ambiguity refers to where the rule itself doesn't syntactically indicate whether to apply a conjunction or disjunction, and world knowledge is required to infer the rule.

H Further details on Follow-up Question Generation Modelling

Table 9 details all the results for all the the models considered for follow-up question generation.

First Sent. Return the first sentence of the rule text

Random Sent. Return a random sentence from the rule text

SurfaceLR A simple binary logistic model, which was trained to predict whether or not a given sentence in a rule text had the highest trigram overlap with the target follow-up question, using a bag of words feature set, augmented with 3 very simple engineered features (the number of sentences in the rule text, the number of tokens in the sentence and the position of the sentence in the rule text)

Sequence Tag A simple neural model consisting of a learnt word embedding followed by an LSTM. Each word in the rule text is classified as either in or out of the subsequence to return using an I/O sequence tagging scheme.

H.1 Further details on neural models for question generation

Table 10 details what the inputs and outputs of the neural models should be.

The NMT-Copy model follows an encoder-decoder architecture. The encoder is an LSTM. The decoder is a GRU equipped with a copy mechanism, with an attention mechanism over the encoder outputs and an additional attention over the encoder outputs with respect to the previously copied token. We achieved best results by limiting the model's generator vocabulary to only very common interrogative words. We train with a 50:50 teacher-forcing / greedy decoding ratio. At test time we greedily sample the next word to generate, but prevent repeated tokens being generated by sampling the second highest scoring token if the highest would result in a repeat.

In order to frame the task as a span extraction task, a simple method of mapping a follow-up question onto a span in the rule text was employed. The longest common subsequence of tokens between the rule text and follow-up question was found, and if the subsequence length was greater than a certain threshold, the target span was generated by increasing the length of the subsequence so that it matched the length of the follow-up question. These spans were then used to supervise the training of the BiDAF and sequence tagger models.

I Evaluating Utility of CMR

In order to evaluate the utility of conversational machine reading, we run a user study that compares CMR with the scenario when such an agent is not available, i.e. the user has to read the rule text, the question, and the scenario, and determine for themselves whether the answer to the question is "Yes" or "No". On the other hand, with the agent, the user does not read the rule text, instead only responds to follow-up questions with a "Yes" or "No", based on the scenario text and world knowledge.

We carry out a user study with 100 randomly selected scenarios and questions, and elicit annotation from 5 workers for each. As these instances are from the CMR dataset, the quality is fairly high, and thus we have access to the *gold* answers and follow-ups questions for all possible responses by the users. This allows us to evaluate the accuracy of the users in answering the question, the primary objective of any QA system. We also track a number of other metrics, such as the time taken by the users to reach the conclusion.

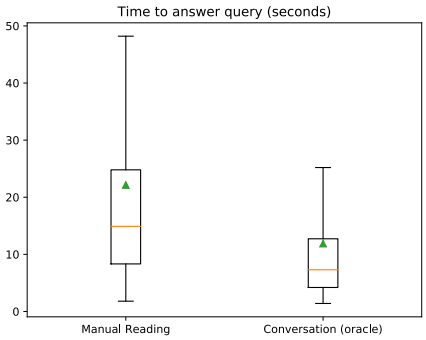
In Figure 10a, we see that the users that have access to the conversational agent are almost twice as fast the users that need to read the rule text. This demonstrates that even though the users with the conversational agent have to answer more questions (as many as the followup questions), they are able to understand and apply the knowledge more quickly. Further, in Figure 10b, we see that users with access to the conversational agents are *much more* accurate than ones without, demonstrating that an accurate conversational agent can have a considerable impact on efficiency.

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Random Sent.	0.302	0.228	0.197	0.179
First Sent.	0.221	0.144	0.119	0.106
Last Sent.	0.314	0.247	0.217	0.197
Surface LR	0.293	0.233	0.205	0.186
NMT-Copy	0.339	0.206	0.139	0.102
Sequence Tag	0.212	0.151	0.126	0.110
BiDAF	0.450	0.375	0.338	0.312
Rule-based	**0.533**	**0.437**	**0.379**	**0.344**

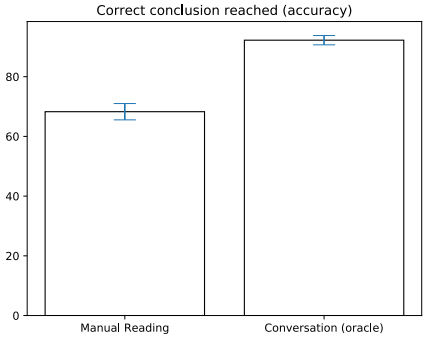
Table 9: All results of the baseline models on follow-up question generation.

Model	Input	Output
NMT-Copy	$r \mid\mid q \mid\mid f_1 \; ? \; a_a \mid\mid \ldots \mid\mid f_m \; ? \; a_m$	f_{m+1}
Sequence Tag	$r \mid\mid q \mid\mid f_1 \; ? \; a_a \mid\mid \ldots \mid\mid f_m \; ? \; a_m$	Span corresponding to follow-up question.
BiDAF	Question: $q \mid\mid f_1 \; ? \; a_a \mid\mid \ldots \mid\mid f_m \; ? \; a_m$ Context : r	Span corresponding to follow-up question.

Table 10: Inputs and outputs of neural models for question generation.



(a) Time taken to reach conclusion



(b) Accuracy of the conclusion reached

Figure 10: **Utility of CMR** Evaluation via a user study demonstrating that users with an accurate conversational agent are not only reach conclusions much faster than ones that have to read the rule text, but also that the conclusions reached are correct much more often.