1 Inter-annotator Agreement

We report the inter-annotator agreement measured by Randolph's κ (Randolph 2005) in Table 1. It can be seen that *Grammatical* and *Relevant* have high agreement as they are easy to judge. *New Info* has lower agreement possibly because it is harder to decide. For the example in Table 2, the question "what is the color of the chair ?" may not be annotated as repetitive as the word "color" doesn't appear in the context, though it is actually covered by the specific value "blue grey". *Logical* and *Specific* have the lowest degree of agreement as they are more subjective criteria. According to the table suggested by Landis and Koch (1977), all the criteria achieved at least moderate agreement.

Criteria	Agreement
Grammatical[0-1]	0.933
Relevant[0-1]	0.853
Logical[0-1]	0.659
New Info[0-1]	0.701
Specific[0-4]	0.546

Table 1: Inter-annotator Agreement measured by Randolph's κ (Randolph 2005)

2 Group-level Qualitative Analysis

We provide a group-level evaluation example in Table 2. We can see that the diversity of MLE is very limited (it gets *#Useful* of only 1, though all 3 questions are valid), and it produces highly generic question. The generations are more diverse for hMup. However, we find that a certain expert of hMup has a style of long and illogical generation, like the second one demonstrated here. (It's abnormal to put chairs *on* a table, and the text is not coherent as it doesn't use a pronoun in the second sentence.) This significantly harms hMup's group-level performance (Table 4) compared to its best single model (Table 3). KPCNet(cluster) produces a diverse and specific generation, and we can clearly see the effect of keyword in its generation.

For the group-level evaluation on `Home & Kitchen`, we also studied the system-level Pearson correlation between the automatic metrics and human judgements. Pairwise-BLEU has a correlation of 0.915 with *#Redundant* ($p < 0.01$), -0.835 with *#Useful* ($p < 0.05$). Avg BLEU is shown only correlates well with *Logical* (correlation: 0.849, $p < 0.05$).

3 Ablation Test

Below we describe the ablation test to check the influence of the components and hyperparameters of the model. These tests are all conducted on the `Home & Kitchen` dataset.

3.1 Additional Metrics

To evaluate the quality of our keyword predictor and keyword bridge, we propose these additional automatic metrics:

P@5 Since the number of keywords in ground truth questions are different across each sample. We take the top 5 keywords with the highest predicted probability as selected keyword set Z^s, and calculates precision@5 by:

$$P@5 = \frac{|Z^s \cap Z^T|}{5} \quad (1)$$

where Z^T is the union of keywords extracted from all ground truth questions of a sample.

Response Rate which is the proportion of conditioned keywords that appears in the corresponding generation, and we report the macro average on all the records. We use this to evaluate the controllability of the keyword conditions.

We also report the average generation length(**Length**) as it is related to almost all metrics proposed above, but neither long or short generation should be considered an indicator of good performance.

3.2 Ablation Factors

These are many important factors and parameters in our model. So we divide the ablation test into 2 logical parts: one for keyword predictor (and the effect of data cleaning on it), and another for keyword bridge.

The ablation factors for keyword predictor are as follows (abbreviated for readability):

- **E**: End2end training of keyword predictor with other component. The training objective is a weighted sum of the 2 objectives (Equation 2 & 3).
- **S**: Separate training, first train predictor, and then freeze its parameters to train other parts.
- **H**: Hard label fed to bridge instead of masked soft logits. The label can be provided from ground truth in training and is decided with threshold filtering in inference. If this setting works well, we can then completely separate the parameters of predictor from other parts.
- **C**: Cleaned dataset.

The ablation factors for keyword bridge are:

- **NE**: No encoder feature fed back to encoder
- **ND**: No decoder feature fed to decoder
- **Dropout**: We add a dropout layer for the unmasked keywords logits before it passes the latter transformation. Due to the nature of dropout, this part may help ease the noise introduced by the error of keyword predictor. And we study the effect of the strength of this layer.

3.3 Results

The ablation test result for data and keyword predictor at individual-level is shown in Table 3. The setting for keyword bridge is fixed: dropout=0.3, both encoder and decoder feature are used. After data cleaning(C), *P@5* dropped because of the reduction of the number of ground-truth keywords. The decreasing of *Distinct-3* and *Length* shows the effect of irrelevant part removing. The improvement on *BLEU* and

product	homelegance 2588s accent dining chair, blue grey, set of 2		
system (#Useful)	generation group	specific	problem
ref (3)	can any of the recent reviewers confirm the seat height ? i see the question was posted in april ...	2	
	would u please send me the box dimensions (when buy in a set of 2) and the weight ?	3	
	can someone please tell me the depth of the chair seat from the end of the curved back to the end of the seat ?	3	
MLE (1)	what is the seat height ?	2	
	what are the dimensions of the chair ?	2	
	what are the dimensions ?	1	
hMup (1)	what is the weight limit for the chair ?	2	
	i have a table that is a [UNK]. will this chair be able to fit on a table ?	2	illogical
	is this a set of 2 chairs or just one ?	2	repetitive
KPCNet (2)	what is the **color** of the **chair** ?	2	repetitive
	what are the **dimensions** of the **seat** ?	2	
	what is the **weight** limit ?	2	

Table 2: Example generation group and the human judgements for each system. Here we use KPCNet to stand for KPCNet(cluster) for brevity, and the responded keywords of KPCNet are highlighted.

	Distinct-3	BLEU	P@5	Response	Length
KPCNet(C, S)	0.1530	17.77	0.472	0.395	7.263
-C	0.1651	15.88	0.510	0.350	7.517
-S, +E	0.1200	9.04	0.217	0.500	7.656
+H	0.2997	12.85	0.472	0.657	9.171

Table 3: Ablation test results on `Home & Kitchen` for data and keyword predictor at individual-level. The first line is final adopted setting.

Response indicates the overall benefits brought by the cleaning. End2end training(-S, +E) leads to significant performance degradation on all metrics except slight increase on *Response*. The possible reason is that keyword prediction skews highly towards frequent keywords under this condition. Finally, feeding hard label instead of logits also produce worse result. We can see from the extremely high *Response* and *Length* that this setting suffers severely from over-generation of keywords: model generates illogical long questions to contain as much keywords as possible. We hypothesize that the soft logits can reflect subtle difference on the importance of each conditioned keyword and thus can lead to more robust performance. Moreover, we can achieve a *P@5* of 0.628 with one group of group truth keywords, as compared to 0.472 of the current model, which shows a huge room for improvement of the keyword predictor.

The ablation test result for keyword bridge at individual-level is shown in Table 4. The setting for keyword predictor is fixed as KPCNet(C, S). We can clearly witness the trend that the higher dropout, the higher controllability keywords will have over generation (Response). As a result, the behavior of KPCNet will be more and more like MLE when dropout grows, with lower generation length, lower keyword response and higher BLEU. We speculate that the dropout imposed on the keywords logits to be masked forces the model to make prediction with incomplete keyword set. Therefore, proper level of dropout can make the model robust to the noise introduced by keyword predictor. Furthermore, the ablation of either encoder bridge or decoder bridge would harm BLEU, response and length, which proved the effect of KPCNet's double-bridge design to guide the generation via attention between the two sides.

We also conducted human evaluation for different value of dropout (Table 5), and found that lower dropout trades logicality for new information. We selected Dropout=0.3 as the final setting for its good balance of all metrics.

4 Experimental Details

4.1 Data Cleaning

The following steps are enforced to remove noises as well as remove unhelpful parts for the CQGen task in the original data:

Fixing Unescaped HTML characters We noticed that there are unescaped HTML special characters in both context and the question. (e.g. "does it slice like zucchini **& amp ;** cucumbers?" is changed to "does it slice like zucchini **&** cucumbers?")

Remove non-question parts Sometimes there are declarative sentences following the question, which is not the focus of our task. We thus removed them. (e.g. For "where is this product made ? i contacted customer service and the representative was uninformed and could not offer any information ." We will remove the second sentence.)

Remove noise questions Some questions contain the comparison between 2 specific entities, which is unlikely to be tackled by our model, so we dropped them. And some questions are too universal ("Does it ship to Canada?"). We consider them as noise and also dropped them.

Note that the data cleaning was only imposed on the training set and the validation set. We preserve exactly the same test set as Rao and Daumé III (2019) for fair comparison.

4.2 Hyperparameters and other settings

For all models, we set the max length of context to be 100, question to be 20. For all variants of KPCNet, we use 2-layer GRU (Cho et al. 2014) with 100 hidden units for both the encoder and decoder. We use a learning rate of 0.0003 to train at most 60 epochs. For MLE, the model structure and parameters are identical to KPCNet, and we follow the setting of Rao and Daumé III (2019), using dropout=0.5, learning rate=0.0001 to train 100 epochs. To improve the generation quality, we block bigrams from appearing more than once, and also forbid 2 same words to appear within 3 steps. For sampling-based keyword selection, we sampled 3 keywords from top-K top-p filtered keywords distribution with $K = 6, p = 0.9$ for 2 times. For clustering-based keyword selection, we produce 2 clusters from the top 6 predicted keywords. For hMup, we use the implementation in fairseq[1]. The architecture is set to 2-layer LSTM (Hochreiter and Schmidhuber 1997) with 100 hidden units, and other settings are identical to KPCNet for fair comparison. The threshold α for the default keyword selection method of KPCNet is manually tuned within range [0.05, 0.1]. The dropout strength is shared among all components of KPCNet and is manually tuned within range [0.2, 0.5]. MLE and KPCNet is implemented in PyTorch. For all manually tuned hyperparameters, we fix all other hyperparameters and random search for value within given range that can achieve the best BLEU on our validation set. The models are trained on a Ubuntu 18.04.4 LTS server with one NVIDIA GeForce RTX 2080 Ti.

For `Home & Kitchen` dataset, all models are operated on 200D word embeddings borrowed from Rao and Daumé III (2019), which are pretrained from in-domain data with Glove (Pennington, Socher, and Manning 2014) and are frozen during training, except for hMup, which uses unique embedding to distinguish between experts and thus the embeddings are trained from scratch. The selected threshold α is 0.07, after 3 trials, and the selected dropout is 0.3 after 4 trials.

For `Office` dataset, all models are operated on 200D word embeddings that we pretrained from in-domain data with Word2vec(Mikolov et al. 2013) in gensim[2], except for hMup. The selected threshold α is 0.07, after 3 trials, and the dropout is initially selected as 0.3 based on the result of `Home & Kitchen`.

For hypothesis test in Table 4, we use `proportions_ztest` of scipy for the first 3 columns

[1] https://github.com/pytorch/fairseq/blob/master/examples/translation_moe

[2] https://radimrehurek.com/gensim/models/word2vec.html

	Distinct-3	BLEU	Response	length
Dropout = 0.2	0.1729	17.11	0.452	7.164
Dropout = 0.3	0.1530	17.77	0.395	7.263
Dropout = 0.4	0.1302	18.33	0.353	6.947
Dropout = 0.4, NE	0.1504	18.19	0.341	6.782
Dropout = 0.4, ND	0.1219	17.47	0.317	6.525
Dropout = 0.5	0.1177	18.53	0.319	6.662

Table 4: Ablation test results for keyword bridge at individual-level on `Home & Kitchen`.

	Grammatical[0-1]	Relevant[0-1]	Logical[0-1]	New Info[0-1]	Specific[0-4]
KPCNet	**0.99**	**0.99**	**0.95**	0.80	1.81
KPCNet(filter)	**0.99**	**0.99**	0.94	0.85	**1.84**
KPCNet	0.98	0.97	0.88	0.84	1.77
KPCNet(filter)	0.98	0.97	0.89	**0.88**	1.80

Table 5: Comparison between KPCNet with Dropout=0.3 (upper half) and Dropout=0.2 (lower half) with individual-level human judgements on 100 sample products from `Home & Kitchen`

whose range is binary, and `ttest_rel` for the other 3 columns. The procedure we assign the underline are: First, we underline the best number at each column. Then we run hypothesis test against every other number. If the difference is not significant, we also underline it, otherwise we don't underline it.

References

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* .

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8): 1735–1780.

Landis, J. R.; and Koch, G. G. 1977. The measurement of observer agreement for categorical data. *biometrics* 159–174.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111–3119.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Randolph, J. J. 2005. Free-Marginal Multirater Kappa (multirater K [free]): An Alternative to Fleiss' Fixed-Marginal Multirater Kappa. *Online submission* .

Rao, S.; and Daumé III, H. 2019. Answer-based Adversarial Training for Generating Clarification Questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 143–155.