Learning to Diversify for Product Question Generation

Haggai Roitman, Yotam Eshel, Alexander Nus,
Eliyahu Kiperwasser
{hroitman,yeshel,anus,ekiperwasser}@ebay.com
eBay
Netanya, Israel

Uriel Singer*
urielsinger@gmail.com
Technion
Haifa, Israel

ABSTRACT

We address the product question generation task. For a given product description, our goal is to generate questions that reflect potential user information needs that are either missing or not well covered in the description. Moreover, we wish to cover diverse user information needs that may span a multitude of product types. To this end, we first show how the T5 pre-trained Transformer encoder-decoder model can be fine-tuned for the task. Yet, while the T5 generated questions have a reasonable quality compared to the state-of-the-art method for the task (KPCNet), many of such questions are still too general, resulting in a sub-optimal global question diversity. As an alternative, we propose a novel learning-to-diversify (LTD) fine-tuning approach that allows to enrich the language learned by the underlying Transformer model. Our empirical evaluation shows that, using our approach significantly improves the global diversity of the underlying Transformer model, while preserves, as much as possible, its generation relevance.

ACM Reference Format:
Haggai Roitman, Yotam Eshel, Alexander Nus, Eliyahu Kiperwasser and Uriel Singer. 2022. Learning to Diversify for Product Question Generation. In *Proceedings of First Content Understanding and Generation for E-commerce Workshop, KDD'2022 (EcomGen2022).* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

E-Commerce is fast-expanding, with a never-ending requirement to offer personalized shopping experiences to users. Product descriptions on E-Commerce websites, such as Amazon, eBay and Shopify, serve as an important knowledge source to potential buyers for making purchase decisions. Product descriptions strive to be as informative and accurate as possible, trying to satisfy a variety of user information needs. In reality, creating product descriptions that can satisfy any possible information need is extremely difficult, as it is hard to anticipate in advance the full range of such needs. The gap between a buyer's information need and the information available in a product description, usually requires the buyer to contact the seller directly with clarification questions or to forfeit her purchase intent, which leads to an undesirable churn.

Table 1: Motivating example: questions generated by three different generative models for two products from the *Home & Kitchen* category

Product	*tuft & needle five handcrafted mattress (twin)*	
KPCNet	will this mattress fit a mattress ? will this mattress fit a queen mattress ?	
T5	what is the warranty on this mattress ? what are the dimensions of this item ?	
T5+LTD	what is the warranty on this mattress ? does it come with a cover to protect the mattress from spills ?	
Product	*gibson couture bands 16-piece dinnerware set, blue and cream*	
KPCNet	are these plates made in the usa ? what are the dimensions of the set ?	
T5	what is the diameter of the dinner plate ? what is the size of the bowls ?	
T5+LTD	are they dishwasher safe ? where is this product made ?	

In this work, we aim to mitigate such gaps by automatically generating product clarification questions to be recommended to sellers. Such a recommendation can take place, for example, already during the product's listing process once the seller provides the product's description. As a result, the seller may revise the product description with missing details. Automatically generated questions should be as **relevant** and **diverse** as possible, covering a multitude of informative aspects that are specific to the product and its usage.

Generating questions that are both relevant and diverse with respect to a specific product is a challenging task. This becomes even more challenging when facing a wide range of information needs over a multitude of product types [25]. As an illustrative example, Table 1 shows two products (only product titles and their images are provided for brevity) and the top-2 questions that were generated by three state-of-the-art generative models: KPCNet [25], T5 [12] and T5+LTD – our proposed solution. Ignoring obvious mistakes such as illogical questions (e.g., *will this mattress fit a mattress ?*), we can observe that, by trying to cover as many product types as possible, some models may "prefer" to generate questions that are too general, yet still relevant in a way (e.g., *what are the dimensions of <product>?*). As this example demonstrates, one may wish to expand the range of questions that can be generated over a given products collection (*global diversity*), while still preserving the ability to generate questions that are both relevant and diverse to a specific product in the collection (*local diversity*).

*Work done while being an intern in ebay.

EcomGen2022, August 14, 2022, Washington, USA
© 2022 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Trying to address the challenge, in this work, we propose a novel *learning-to-diversify* (LTD) fine-tuning approach for product question generation using Transformers [18]. To this end, using a bi-branch network architecture, we fine-tune the underlying pre-trained T5 Transformer encoder-decoder model by training it in a pairwise-way with multiple question-pairs per product. Our goal is to maximize the generation likelihood of each pair of questions, while at the same time, minimize their semantic similarity. The semantic similarity between a pair of questions is measured with respect to the latent query representations learned by the Transformer's decoder. Applying our approach on the underlying Transformer model requires no further change during inference time. Utilizing such a learning approach allows to enhance the underlying Transformer model's ability to generate diverse questions which cover a much broader range of product information needs, resulting in an increase in its global (question generation) diversity. This is done, while still preserving (as much as possible) the underlying Transformer model's ability to generate diverse questions that are relevant to a specific product in the collection, hence, preserving its local (question generation) diversity.

Using product descriptions and questions from different product categories on Amazon, we demonstrate that, the quality of questions that are generated using our fine-tuning approach (LTD) is better than of those generated by several alternative models, including the underlying pre-trained T5 Transformer model when it is fine-tuned in the "traditional" way.

The rest of this paper is organized as follows. We discuss related works in Section 2 and present our learning framework in Section 3. We report our evaluation in Section 4 and conclude in Section 5.

2 RELATED WORK

We review works primarily related to either product or diversified question generation tasks. A more general overview on the question generation task in NLP can be found in [23].

2.1 Product question generation

The product question generation task is a relatively new task. A common approach, is to model the generation process as a sequence-to-sequence (seq2seq) setting, using the product's description as the source text to be encoded and the required question as the target text to be decoded [23]. Yet, vanilla recurrent-neural networks that are applied to the task suffer from common problems such as unknown words and difficulty to control the specificity and diversity of generated questions [7]. Xiao at el. [21] have handled unknown words (yet not diversity) using a pointer-generator network. Zhang et al. [25] have proposed KPCNet – a seq2seq model that attended on selected product description keywords to improve question specificity. Several other works [13, 20, 22] have utilized adversarial learning (e.g., GANs) to improve question generation "quality" by using an additional discriminator model. Yet, training such a discriminator requires additional labeled data with question answers. Wang et al. [20] have further suggested to train the discriminator with question pairs, consistenting of a true question and a negatively sampled one. Yet, the generator in [20] heavily depends on the availability of auxiliary data such as product properties and user interest aspects. Finally, Majumder et al. [9] have utilized global product knowledge to predict missing aspects.

2.2 Diversified question generation

Enhancing the diversity of text generation, and questions in particular, was the aim of several previous works [2–4, 7, 15, 16]. A common approach to diversify the generated questions is to apply diversification methods during model inference [4]. Common methods include: diverse beam-search (DBS), top-p and/or top-k sampling and post-generation analysis (e.g., clustering [25], specificity classification [2]). Yet, such inference methods strongly depend on the language learning capacity of the underlying trained model. An alternative approach is, therefore, to allow the model to "explore" more during its training phase [7]. To the best of our knowledge, only few related works have focused on such an approach. Shen et al. [16] have trained a mixture of experts model to learn different generation styles. Shao et al. [15] have proposed Apex – a Conditional Variational Auto-encoder for product description generation from few keywords. The trade-off between accuracy and diversity was controlled by setting a bound on the KL-divergence loss. Cho et al. [3] have employed contrastive learning for question generation over multiple documents. To this end, their model was trained on triplets containing a single training question with a pair of positive and negative document sets. The generator's goal was to generate questions that are only grounded in the positive documents.

2.3 Main differences

The main goal of our work is to improve the global diversity of questions generated by an underlying pre-trained Transformer [18] model. Most existing works have fine-tuned pre-trained Transformer models with the primary objective of maximizing question generation likelihood; while diversity was commonly dealt only as a secondary objective during inference time [8, 23]. Compared to [9], which have utilized the Transformer model for the same task, we do not use any auxiliary data. Finally, our method is eminently different from existing contrastive learning methods [3], as in our case there are only positive examples. To the best of our knowledge, we are not aware of any similar work that has fine-tuned pre-trained Transformer models for enhanced diversity as we do.

3 FRAMEWORK

We first formally define the product question generation (PQG) task (Section 3.1). We then shortly discuss how the T5 pre-trained Transformer encoder-decoder model [12] can be fine-tuned for this task (Section 3.2). Finally, we introduce our alternative learning-to-diversify (LTD) fine-tuning approach (Section 3.3).

3.1 Product Question Generation Task

For a given product description text sequence, termed hereinafter as "context", $c = (x_1, x_2, \dots, x_n)$, the goal of the PQG task is to generate a question text sequence $q = (y_1, y_2, \dots, y_m, ?)$. The generated question q should be informative enough to clarify details related to the product (e.g., an inquiry about a specific product aspect such as its color or size, usage, compatibility, etc) which are not (fully) described in c. Commonly, several questions $Q_c = \{q_1, q_2, \dots, q_k\}$

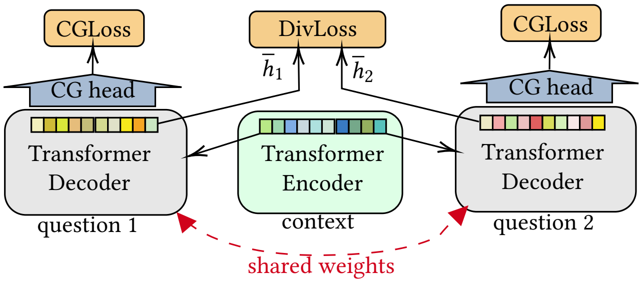

Figure 1: Learning-to-diversify for PQG task

may be generated for a given context c. The questions-set Q_c should be as diverse as possible, covering different potential user information needs regarding the product. In that case, we say that Q_c is *locally diverse*. For a given products collection $C = \{c_1, c_2, \ldots, c_g\}$, we further wish that the corresponding generated questions superset $Q_C = \bigcup_{c \in C} Q_c$ would be as diverse as possible, covering a wide range of information needs over all products in the collection C. In that case, we say that Q_C is *globally diverse*.

3.2 Fine-tuning the T5 model for the PQG task

In this work, we utilize the T5 [12] pre-trained Transformer [18] encoder-decoder model and fine-tune it for the PQG task. "Traditionally", model fine-tuning is implemented as a *conditional generation* (seq2seq) task [23]. Formally, for a given training sample (c, q), the goal is to maximize the generation likelihood:

$$p(q|c, \theta) = \prod_{t=1}^{m} p(y_t | q_{<t}, c, \theta), \tag{1}$$

where $q_{<t}$ denotes the question words that were generated up to step t and θ represents the Transformer model parameters.

Next, we shortly describe how the Transformer encoder-decoder model can be utilized for this task. Before we move on, it is important to mention that, several Transformer "block" layers can be stacked together to increase the model's learnability [18]. For full technical details on the Transformer model, we kindly refer the reader to [18]. Given context c, the Transformer encoder first encodes it into a latent representation $h_e = Trans_{enc}(c)$. At step t, the Transformer decoder block at layer $l \in \{1, 2, \ldots, L\}$ attends both on h_e and the output of the previous layer $h_d^{(l-1)}$, and then outputs a representation $h_d^{(l)} = Trans_{dec}(h_e, h_d^{(l-1)})$; with $h_d^{(0)} = q_{<t}$. The next query word y_t is then generated using a conditional generation "head" (CG-head[1]), which transforms $h_d^{(L)}$ into a distribution over the Transformer's vocabulary and assigns y_t as the word with the highest likelihood. Training is usually implemented using a *teacher-forcing* approach [23], where at each step t, the training question words up to step t are used instead of $q_{<t}$. To simplify notation, from now on, we shall abbreviate h_d as h.

3.3 Learning to Diversify

We next propose an alternative (fine-tuning) approach whose primary goal is to enhance the overall (global) diversity of questions generated by the underlying Transformer model.

Our approach is built on the hypothesis, which is empirically verified later on using the T5 model, that, pre-trained Transformer models for the PQG task may tend to learn common (general) questions that appear in the training set in the expense of more rare ones. Therefore, we wish to improve the "exploration" capability of the underlying Transformer model by allowing it to learn a more flexible language model that results in a generation of more (globally) diverse questions.

Our proposed learning-to-diversify (LTD) fine-tuning approach is based on a *bi-branch* network architecture and is illustrated in Figure 1 (assuming $L = 1$ for simplicity) and works as follows. We train the underlying Transformer model on triplets (c, q_1, q_2), where q_1 and q_2 are a pair of different target questions that the model needs to generate for a given input context c. This part is simply implemented as before, where for each question q_i $(i \in \{1, 2\})$, we generate the next word y_t^i based on the Transformer decoder representation h_i, respectively. Let $CGLoss_t^i$ be the overall corresponding conditional generation (CG) loss[2] incurred by generating the question words $\hat{q}_i = (y_1^i, \ldots, y_{m_i}^i)$.

Next, we "infuse" exploration to the underlying model by encouraging it to generate two questions that are eminently semantically different from each other. To this end, following the **teacher-forcing** approach, for each question q_i we set $h_i^{(0)} = q_i$ and obtain its representations $h_i^{(l)}$ over all the Transformer decoder layers. Here we note that, the key idea behind such an approach, is to obtain a given question's representation by the decoder assuming that the model **has correctly generated it**. Our goal is therefore, to allow a backward feedback to the model based on how different are the two questions representations are. The overall difference between the two questions representations by the decoder is measured according to the cosine loss term between the representations that were obtained by the Transformer decoder with L blocks (stack): $DivLoss^{1,2} = \frac{1}{L}\sum_{l=1}^{L} cosine(\bar{h}_1^{(l)}, \bar{h}_2^{(l)})$, where $\bar{h}_i^{(l)}$ is calculated using *mean-pooling* over the sequence dimension[3] of $h_i^{(l)}; i \in \{1, 2\}$, respectively.

Finally, we use the diversity loss as a regularization term for the two CG losses, as follows:

$$Loss = CGLoss^1 + CGLoss^2 + \lambda \cdot DivLoss^{1,2}, \tag{2}$$

where the hyperparameter $\lambda > 0$ controls to what extent we wish the model to explore towards diversification.

4 EVALUATION

4.1 Experimental Setup

4.1.1 Datasets. We summarize the details of the datasets that we use for our evaluation in Table 2. All datasets are based on products sold on Amazon and questions that were asked by Amazon

[1] Implemented using a feed-forward (FF) layer, followed by a softmax operation [18].

[2] Implemented as cross-entropy loss [23].

[3] $h_i^{(l)}$ is a matrix defined over the input (sequence) and embedding dimensions [18].

Table 2: Datasets used for the evaluation

Category	#Products	#Questions	Train	Validation	Test
Home & Kitchen	23,859	145,536	19,119	2,435	2,305
Office Products	2,731	13,775	2,190	285	256
Sports & Outdoors	8,398	54,383	6,664	834	835
Electronics	23,900	166,182	19,108	2,389	2,389

buyers [10]. Following previous works [13, 25], we use product-question pairs sampled from the *Home & Kitchen* and the *Office Products* categories of the Amazon dataset. We use the pre-processed dataset version of [25] for both categories. We further extend our evaluation with product-question pairs that are sampled from two additional categories, namely: *Sports & Outdoors* and *Electronics*. Following [25], each product context consists of the concatenation of the product title and description. On average, on each dataset, each product context has about 3-10 questions. On each dataset, we use about 80% of the products for training, 10% for validation (tuning) and the last 10% for testing.

4.1.2 Baselines. Our first line of baselines are those that were previously evaluated in [25]. This includes: **MLE** – a vanilla seq2seq model [25], **hMup** [16] – a mixture of experts model, and **KPC-Net** [25] – the current state-of-the-art method for the PQG task. For a fair comparison, we use the best baseline results[4] that are provided by [25].

Since we apply our learning-to-diversify (LTD) fine-tuning approach on the pre-trained T5 [12] model, we further evaluate the same pre-trained model when it is fine-tuned using the "traditional" approach (Section 3.2), denoted **T5**. Finally, we evaluate **T5+LTD**, the same T5 pre-trained model fine-tuned with our LTD approach.

4.1.3 Implementation and Training. We implement both fine-tuning approaches with pytorch and huggingface[5] pre-trained T5 model ("t5-base"). We use the Adam [5] optimizer, with a learning rate of 10^{-4}, a batch size of 8, trained for 3 epochs (saving the best checkpoint using the validation-set) with a machine with 4 GPUs. For a fair comparison, we fix the inference of all Transformer-based models (i.e., T5 and T5+LTD) and use the diversity beam-search (DBS) [19] method; which gives the best inference quality (on the validation-set) compared to other alternative inference methods [4]. We tune (on the validation-set) $\lambda \in (0, 1]$, with $\lambda = 0.1$ derived as the "best" hyperparmeter value choice over all categories. Following [25], using the validation-set, we set the number of beam groups to 3 (with 6 total questions generated per product), length penalty of 1.0, diversity penalty of 5.0 and non-repeat of bi-grams.

Table 3: Comparison of question generation relevance obtained by the various baselines (*Home & Kitchen category*)

	Lexical Relevance			Semantic Relevance	
	BLEU$\uparrow$	Avg-BLEU$\uparrow$	METEOR$\uparrow$	BERTScore$\uparrow$	Avg-BERTScore$\uparrow$
MLE	18.1	16.9	14.9	30.9	30.9
hMup	17.8	9.9	15.4	23.5	28.4
KPCNet	17.8	16.2	16.2	32.3	31.6
T5	19.2	17.1	**16.4**	32.1	31.6
T5+LTD	**20.1**	**17.2**	16.3	**32.9**	**31.8**

4.1.4 Metrics. We evaluate both the **relevance** and the **diversity** of the generated questions. Following [25], for each product, we evaluate the top-3 generated questions. We measure the generated questions relevance along two main dimensions, namely: **lexical** (based on surface-level lexical overlap) and **semantic** (based on word-context). Following [25], we measure lexical relevance using the (top-1 question) **BLEU** [11], (top-3 questions) **Avg-BLEU** [25] and (top-1 question) **METEOR** [1] metrics. Following [24], we measure semantic relevance using the **BERTScore**[6] (top-1 question) and **Avg-BERTScore** (top-3 questions) metrics.

We measure the diversity of generated questions both **locally** (per-single product) and **globally** (per products-collection). Similar to the relevance metrics, we use both lexical and semantic measures. Following [25], we measure the lexical local diversity according to Pairwise-BLEU (abbreviated as **PW-BLEU** in our tables) (top-3 questions). This is an extension of the Self-BLEU metric [17], having every time one question (out of top-3) being considered as the "hypothesis" and the rest as "references" [25]. As a semantic alternative of this measure, we further calculate Pairwise-BERTScore (abbreviated as **PW-BERTScore** in our tables), where we replace BLEU with BERTScore. Here we note that, since the goal is to have three questions per product that are different from each other, **lower** PW-BLEU and PW-BERTScore values translate to better local diversity.

We further measure lexical global diversity of generated questions using the Distinct-N [7] metric (abbreviated as **Dist-N** in our tables); where $N \in \{1, 2, 3\}$ denotes the N-gram size. This metric is calculated by counting (considering all top-1 questions generated for the test-set products) the number of unique N-grams, divided by the total number of N-grams [7]. Since this metric is lexical in its nature (i.e., counts exact words), we further wish to evaluate global diversity using a more semantic measure. Following [6], we use the embedding-based diversity measure (abbreviated as **e-Div** in our tables). To this end, given a collection of question embeddings, for each embedding dimension, we first calculate its radius (the standard-deviation of values in that dimension [6]). The embedding-based diversity (e-Div) is then calculated as the geometric mean of the radius values. Hence, the larger is the radius over each dimension, the more spread is the questions embedding space, and therefore, more (globally) diverse. Finally, we obtain the question embeddings (with 512 dimensions) using a SentenceTransformer [14] encoder dedicated for paraphrasing tasks.

4.2 Results

We now summarize the results of our empirical evaluation. Our goal is to answer the following three main research questions (RQs):

- **RQ1**: How good (both in terms of relevance and diversity) are the questions generated by the T5 baseline compared to those generated by the best previously performing method (KPCNet) for the PQG task?

[4] Baselines code and results are available in: https://github.com/blmoistawinde/KPCNet

[5] https://huggingface.co/docs/transformers/index

[6] https://github.com/Tiiiger/bert_score

Table 4: Comparison of question generation diversity obtained by the various baselines (*Home & Kitchen category*)

	Local Diversity		Global Diversity			
	PW-BLEU↓	PW-BERTScore ↓	Dist-1↑	Dist-2↑	Dist-3↑	e-Div↑
MLE	26.8	61.6	2.7	5.7	7.8	29.7
hMup	**13.4**	**44.5**	2.3	9.6	11.1	26.8
KPCNet	42.8	75.2	3.4	9.6	15.3	29.9
T5	27.1	58.2	4.1	9.2	14.3	29.8
T5+LTD	26.2	54.0	**5.1**	**11.5**	**17.5**	**30.4**

- **RQ2**: How good are the questions generated by T5+LTD compared to T5?
- **RQ3**: How more globally diverse are questions generated by T5+LTD compared to T5?

4.2.1 RQ1: "Traditional" pre-trained T5 model fine-tuning. To answer RQ1, we compare all baselines over the *Home & Kitchen* category, for which the **best** generated questions of the MLE, hMup and KPCNet baselines are publicly available. We report the evaluation results in Table 3 (relevance) and Table 4 (diversity). As we can observe, on relevance, the pre-trained T5 model that is fine-tuned with the traditional approach (see Section 3.2) generates questions that are more lexical relevant (meaning it generates more relevant question words) compared to previously studied baselines (specifically KPCNet – the existing state-of-the-art method). Yet, when it comes to semantic relevance, the same baseline (T5) obtains only competitive question relevance to that of KPCNet. That actually means that, while the model is capable of generating novel (relevant) words (probably due higher vocabulary coverage and better context modeling), the same model does not actually contribute novel questions with new meaning on top of those generated by KPCNet. This empirical outcome serves as a first evidence to our motivation: traditional fine-tuning of the pre-trained T5 model may result in sub-optimal diversity. And indeed, as we can further observe, the T5 baseline has a global diversity that is inferior (in 3 out of the 4 metrics) to that of KPCNet, even though T5 still results in a better local diversity (i.e., lower PW-BLEU[BERTScore] values).

We next further examine the question generation quality (both with respect to the relevance and diversity metrics) obtained by T5+LTD (i.e., the pre-trained T5 model fine-tuned with our alternative approach). For relevance, we can now observe that, T5+LTD obtains better question relevance (except for the METEOR metric for which it has more or less similar performance to T5), both lexically and semantically. This implies that, fine-tuning the pre-trained T5 model using our LTD approach allows it not only to generate novel words that cover more questions, but also new questions that contribute new meaning to the poll of questions that can be asked over the products collection. This is strongly supported by the diversity metrics obtained by T5+LTD compared to the other baselines, and T5 specifically, where the former significantly outperforms the others in all global diversity metrics (both lexical and semantic). T5+LTD further significantly outperforms all other baselines (accept hMup[7]) on the local diversity metrics. This shows that, while is was mainly designed to improve global diversity (over all products), it is also capable of generating diverse questions on a per-product basis. Overall, these first empirical results demonstrate that, T5+LTD is capable of generating more diverse questions, while preserving the relevance of the underlying T5 Transformer model as much as possible.

4.2.2 RQ2: Impact of our learning-to-diversify fine-tuning approach. To answer RQ2, we deepen our comparison between T5 and T5+LTD. We report in Table 5 the results of this comparison, when the two alternatives are trained to generate questions for products in different categories. Therefore, such a comparison provides a better analysis of the robustness of our diversification approach considering a variety of product categories.

We note again that, the LTD approach is applied on the underlying Transformer model **only during training**, while the inference remains completely **unchanged**. Hence, in both fine-tuning alternatives, we start from the **same** pre-trained T5 model and the fine-tuned model is then evaluated on the same grounds.

As we can observe, for all product categories, applying the LTD approach results in a significant boost in global diversity, both lexically and semantically (up to more than 40% for some of the metrics). The LTD approach encourages the underlying Transformer model to learn a much richer language model for the task, which results in a more diverse generation. Moreover, in most cases, the local diversity is also improved (specially semantic diversity which always improves).

Examining the relevance metrics, we can observed that, in the majority of cases, T5+LTD results with a reasonable question generation relevance (for some categories even much better, for some with a relatively slight drop). This serves as another strong empirical evidence that, our LTD approach not only improves the underlying Transformer model diversity (meaning novel questions are being generated), but also preserves, as much as possible, its capability of generating relevant questions.

4.2.3 RQ3: Global diversity topic analysis. To understand better the impact of our LTD approach on the underlying Transformer model's global diversity, we next perform topic analysis on the generated questions. We hypothesize that, using our LTD approach should result in a significant semantic improvement in terms of number of information needs ("topics") that can be learned. To this end, using a SentenceTransformer [14] dedicated for paraphrasing tasks, for each product category, we obtain embeddings for the test-set generated questions of both T5 and T5+LTD. Next, using cosine similarity as the "distance" metric, we obtain question clusters[8] for each alternative and measure the number of clusters obtained for increasing (dissimilarity) thresholds. We report the results in Figure 2, illustrating the relative improvement (green bars) or degradation (red bars) of T5+LTD compared to T5. As we can observe, for all product categories, for almost every dissimilarity threshold, T5+LTD obtains significantly more clusters (a statistically validated result $p < 10^{-4}$). This serves as another strong empirical evidence for the ability of the LTD approach to enrich the language that can be learned by the underlying Transformer model.

[7] This baseline was mainly designed to obtain high local diversity, yet this comes with the expense of a large relevance drop [25].

[8] Clusters obtained with scikit-learn's Agglomerative-Clustering.

Table 5: Comparison of question generation quality between T5 and T5+LTD for different product categories. The percentages bellow the T5+LTD metric values denote the relative improvement/degradation compared to T5.

Category		Lexical Relevance			Semantic Relevance		Local Diversity		Global Diversity			
		BLEU↑	Avg-BLEU↑	METEOR ↑	BERTScore↑	Avg-BERTScore↑	PW-BLEU↓	PW-BERTScore↓	Dist-1↑	Dist-2↑	Dist-3↑	e-Div↑
Home & Kitchen	T5	19.2	17.1	**16.4**	32.1	31.6	27.1	58.2	4.1	9.2	14.3	29.8
	T5+LTD	**20.1** (+4.7%)	**17.2** (+0.6%)	16.3 (-0.6%)	**32.9** (+2.5%)	**31.8** (+0.6%)	**26.2** (+3.4%)	**54.0** (+7.8%)	**5.1** (+24.4%)	**11.5** (+25.0%)	**17.5** (+22.4%)	**30.4** (+2.0%)
Office Products	T5	**16.9**	**13.6**	14.9	**32.2**	**30.9**	28.2	69.6	12.2	21.7	27.9	25.2
	T5+LTD	**16.9**	13.5 (-0.7%)	**15.3** (+2.7%)	31.0 (-3.9%)	29.0 (-6.7%)	**27.2** (+3.7%)	**64.9** (+7.2%)	**14.5** (+18.8%)	**28.1** (+29.5%)	**35.6** (+41.3%)	**25.8** (+2.4%)
Sports & Outdoors	T5	**4.5**	2.5	**10.2**	**23.1**	**22.5**	**32.5**	69.2	7.1	13.9	18.8	25.2
	T5+LTD	4.4 (-2.3%)	**2.6** (+4.0%)	10.1 (-1.0%)	23.0 (-0.4%)	22.4 (-0.4%)	34.2 (-5.2%)	**68.4** (+1.2%)	**8.5** (+19.7%)	**17.1** (+23.0%)	**23.2** (+23.4%)	**25.7** (+2.0%)
Electronics	T5	4.6	**2.8**	9.6	20.5	20.4	**39.8**	72.5	3.6	6.9	9.5	23.1
	T5+LTD	**4.9** (+6.5%)	**2.8**	**10.3** (+7.3%)	**22.3** (+8.8%)	**21.4** (+4.9%)	40.4 (-1.5%)	**71.5** (+1.4%)	**3.7** (+2.8%)	**7.9** (+14.5%)	**11.0** (+15.8%)	**24.0** (+3.9%)

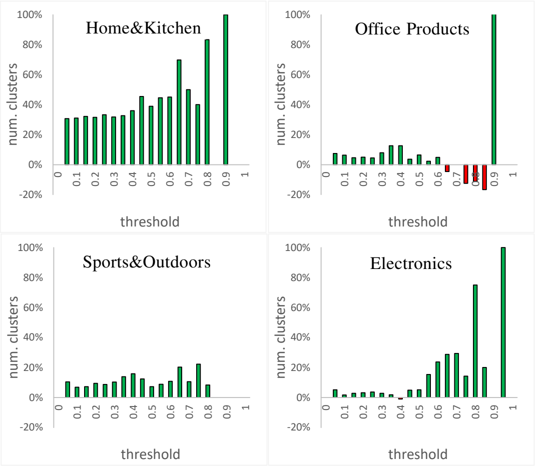

Figure 2: Global diversity topic analysis. The percentages denote the relative improvement (green bars) / degradation (red bars) in the number of clusters per dissimilarity threshold compared to the T5 baseline.

5 SUMMARY

In this work, we have proposed a novel learning-to-diversify (LTD) approach for enhancing the fine-tuning of pre-trained Transformer models for the product question generation task. Using our approach, allows to learn a more globally diverse language model that covers a wider range of user product information needs; this, while preserving the underlying model's generation quality per product in the collection. As future work, we wish to evaluate our approach over other text generation architectures (e.g., Transformer decoder-only models) and tasks.

REFERENCES

[1] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Association for Computational Linguistics, Ann Arbor, Michigan, 65–72. https://aclanthology.org/W05-0909

[2] Yang Trista Cao, Sudha Rao, and Hal Daumé III. 2019. Controlling the Specificity of Clarification Question Generation. In *Proceedings of the 2019 Workshop on Widening NLP*. Association for Computational Linguistics, Florence, Italy, 53–56. https://aclanthology.org/W19-3619

[3] Woon Sang Cho, Yizhe Zhang, Sudha Rao, Asli Celikyilmaz, Chenyan Xiong, Jianfeng Gao, Mengdi Wang, and Bill Dolan. 2021. Contrastive Multi-document Question Generation. In *Proceedings of ACL*. Association for Computational Linguistics, Online, 12–30. https://doi.org/10.18653/v1/2021.eacl-main.2

[4] Daphne Ippolito, Reno Kriz, João Sedoc, Maria Kustikova, and Chris Callison-Burch. 2019. Comparison of Diverse Decoding Methods from Conditional Language Models. In *Proceedings of ACL*. Association for Computational Linguistics, Florence, Italy, 3752–3762. https://doi.org/10.18653/v1/P19-1365

[5] Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR (Poster)*.

[6] Yi-An Lai, Xuan Zhu, Yi Zhang, and Mona Diab. 2020. Diversity, Density, and Homogeneity: Quantitative Characteristic Metrics for Text Collections. In *Proceedings of LERC*. European Language Resources Association, Marseille, France, 1739–1746. https://aclanthology.org/2020.lrec-1.215

[7] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A Diversity-Promoting Objective Function for Neural Conversation Models. In *Proceedings of NACAL-HLT*. Association for Computational Linguistics, San Diego, California, 110–119. https://doi.org/10.18653/v1/N16-1014

[8] Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2022. A Survey of Pretrained Language Models Based Text Generation. arXiv:2201.05273 [cs.CL]

[9] Bodhisattwa Prasad Majumder, Sudha Rao, Michel Galley, and Julian McAuley. 2021. Ask what's missing and what's useful: Improving Clarification Question Generation using Global Knowledge. *arXiv preprint arXiv:2104.06828* (2021).

[10] Julian McAuley and Alex Yang. 2016. Addressing Complex and Subjective Product-Related Queries with Customer Reviews. In *Proceedings of WWW* (Montréal, Québec, Canada) *(WWW '16)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 625–635. https://doi.org/10.1145/2872427.2883044

[11] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL* (Philadelphia, Pennsylvania) *(ACL '02)*. Association for Computational Linguistics, USA, 311–318. https://doi.org/10.3115/1073083.1073135

[12] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67. http://jmlr.org/papers/v21/20-074.html

[13] Sudha Rao and Hal Daumé III. 2019. Answer-based Adversarial Training for Generating Clarification Questions. In *Proceedings of NACAL-HLT*. Association for Computational Linguistics, Minneapolis, Minnesota, 143–155. https://doi.org/10.18653/v1/N19-1013

[14] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arXiv:1908.10084 [cs.CL]

[15] Huajie Shao, Jun Wang, Haohong Lin, Xuezhou Zhang, Aston Zhang, Heng Ji, and Tarek Abdelzaher. 2021. *Controllable and Diverse Text Generation in E-Commerce*. Association for Computing Machinery, New York, NY, USA, 2392–2401. https://doi.org/10.1145/3442381.3449838

[16] Tianxiao Shen, Myle Ott, Michael Auli, and Marc'Aurelio Ranzato. 2019. Mixture Models for Diverse Machine Translation: Tricks of the Trade. arXiv:1902.07816 [cs.CL]

[17] Raphael Shu, Hideki Nakayama, and Kyunghyun Cho. 2019. Generating Diverse Translations with Sentence Codes. In *Proceedings of ACL*. Association for Computational Linguistics, Florence, Italy.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. arXiv:1706.03762 [cs.CL]

[19] Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. 2016. Diverse Beam Search: Decoding Diverse Solutions from Neural Sequence Models. *ArXiv* abs/1610.02424 (2016).

[20] Yongzhen Wang, Kaisong Song, Lidong Bing, and Xiaozhong Liu. 2021. Harvest shopping advice: Neural Question Generation from multiple information sources in E-commerce. *Neurocomputing* 433 (2021), 252–262.

[21] Kang Xiao, Xiabing Zhou, Zhongqing Wang, Xiangyu Duan, and Min Zhang. 2019. Question generation based product information. In *CCF International Conference on Natural Language Processing and Chinese Computing*. Springer, 445–455.

[22] Qian Yu, Lidong Bing, Qiong Zhang, Wai Lam, and Luo Si. 2020. Review-based Question Generation with Adaptive Instance Transfer and Augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 280–290. https://doi.org/10.18653/v1/2020.acl-main.26

[23] Ruqing Zhang, Jiafeng Guo, Lu Chen, Yixing Fan, and Xueqi Cheng. 2021. A Review on Question Generation from Natural Language Text. *ACM Trans. Inf. Syst.* 40, 1, Article 14 (sep 2021), 43 pages. https://doi.org/10.1145/3468889

[24] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SkeHuCVFDr

[25] Zhiling Zhang and Kenny Zhu. 2021. *Diverse and Specific Clarification Question Generation with Keywords*. Association for Computing Machinery, New York, NY, USA, 3501–3511.