Mitigating Entity-based Knowledge Conflicts in Question Answering

Han Cao **Xuhui Liu** **Zixuan Wang**
Yen-Ting Huang **Zhouhang Xie** **Keren Shao**
University of California, San Diego
{h2cao, xul019, ziw055 yeh001, zhx022, k5shao }@ucsd.edu

Abstract

Knowledge-aware sequence to sequence generation tasks such as document question answering and abstract summarization typically requires two types of knowledge: encoded parametric knowledge and retrieved contextual information. Previous work show improper correlation between parametric knowledge and answers in the training set could cause the model ignore input information at test time, resulting in un-desirable model behaviour such as over-stability and hallucination. **In this work**, we argue that hallucination could be mitigated via *explicit correlation* between input source and generated content. We focus on a typical example of hallucination, entity-based knowledge conflicts in question answering, where correlation of entities and their description at training time hinders model behaviour during inference.

1 Introduction

Large pre-trained language models are a common building blocks for a wide array of natural language processing (NLP) tasks. These models are known to encode factual knowledge during training [3, 15]. Such property have been exploited in tasks such as closed-book question answering [29] and commonsense reasoning [41]. However, recent study shows that improperly activated parametric knowledge during inference could cause the model to produce factually inconsistent outputs. Such inconsistency is undesirable in natural language generation (NLG) tasks that requires *grounding* with respect to certain contextual information, affecting performance in various tasks such as table-to-text generation [36], abstract summarization [25], and grounded dialogue systems [9].

A particular challenge in ensuring factual consistency in grounded NLG tasks is that the usually unsatisfied *sufficiency* between the grounding context and gold output. In the ideal world, the input context should always contain enough information for a model to produce the gold output. However, such an assumption is impractical in many tasks. For example, human written gold answers in summarization often contains extrinsic hallucination [25, 8], where extra background knowledge is required for the model to produce the correct output. Similarly, the one-to-many problem in dialogue system [1] were also caused by the insufficiency to accurately predict response given contextual information. Thus, it is crucial that we develop training scheme to constrain model behaviour under such sufficiency constraint.

We study a simple variant of hallucination, entity-based knowledge conflicts [22]. Concretely, recent study find reader model exhibits over-reliance on parametric knowledge, primarily due to information insufficiency caused by an in-perfect retriever during training. In other words, the reader model learns to ignore retrieved documents and rely on parametric knowledge during training, resulting in hallucination at test time. While such behaviour *can* be beneficial, un-alerted hallucination

[1] Where large amounts of answers are equally appropriate given the dialogue history

significantly affects model usability and trustworthiness. We propose two set of orthogonal methods to mitigate undesirable model memorization: gradient based decoding and adapter-based fine tuning. We hope to show that with minimum or no performance sacrifice, our model generate output that is more faithful with respect to input context.

2 Background and Related Works

Hallucination in Text Generation As mentioned in the introduction though, some early studies [12, 37] focused on the potential pitfalls of leveraging standard likelihood maximization-based objectives in training and decoding Natural Language Generation models. They found that an approach that maximizes this potential could lead to deterioration. At the same time, it turns out that these models often produce meaningless text or are not faithful to the source input provided [18, 28]. Researchers have come to call this unwelcome generation a hallucination. [26] Hallucination can lead to potential privacy violations. Carlini et al. [4] demonstrated that language models can be prompted to recover and generate sensitive personal information from the training corpus. Such memorization and recovery of the training corpus is considered a form of hallucination because the model is generating text that is not exist in the source context.

Considering QA models searches external knowledge for information relevant to the question, and generates the answer based on the retrieved information [19], an essential goal is to provide documents-based answers given the question, so that hallucination in the answer will mislead the user and harm performance dramatically.

Controllable Generation Controllable text generation [42, 17, 27] may be one tool to resolve the hallucination problem. Unlike plain text generation, we want the sentences generated by the models (x) to align with specific attributes (a), such as sentiment and content. Works in this area typically deal around the conditional probability $p(x|a)$. While some authors [10] decide to directly model it, some [6, 13] avoid direct modeling through bayes rules or latent variables. For example, Hu et al. [13] uses a latent variable (z) to transform the problem into modeling $p(x|z, a)$, $p(a|x)$ and $p(z|x)$ separately. This tool may allow us to treat the easily hallucinated entities as attributes and consequently train the model not to completely ignore the provided context.

Question Answering Question answering (QA) is a major research direction of NLP. Two of the largest subbranches under QA are visual question answering [2, 23, 40], which uses images as sources of knowledge and context, and text question answering [38, 1], which uses text as the sources. One of major subbranches under text question answering is called open-domain question answering. In this task, given any question, the model is expected to find the answer to it from a giant database of passages, e.g. Wikipedia articles. The state-of-the-art is the retriever-reader approach, as mentioned in Chen et al [5]. With this approach, first, a retriever, such as a dense retriever [16] or simply a BM25 [30], will first retrieve a small set of relevant passage from the given database of passages. A document reader, typically a standard language model like RNN [31] or BERT [7], will then read through these retrieved passages and produce the answer to the question[2].

3 Methods

This section is organized as follows: we first cover definition of knowledge conflicts and a simple data augmentation baseline solution. Then we introduce our projected directions that could potentially lead to better solution to knowledge conflicts. Finally, we report our tentative schedule for the project.

3.1 Problem re-formalization and Model selection

Seq2seq-based big QA models like BART, when accompanied by QA datasets with contexts, become hard to train and require computing resource beyond our current constraints. Hence, to seamlessly handle the datasets we used in our experiments, which typically contain long contexts with more than 150 words, we used pretrained generative language models and finetuned them for QA tasks.

[2] Our current experiments are conducted on a simplified setting, where a language model generates the answer given the *gold* context. See experiment section for details.

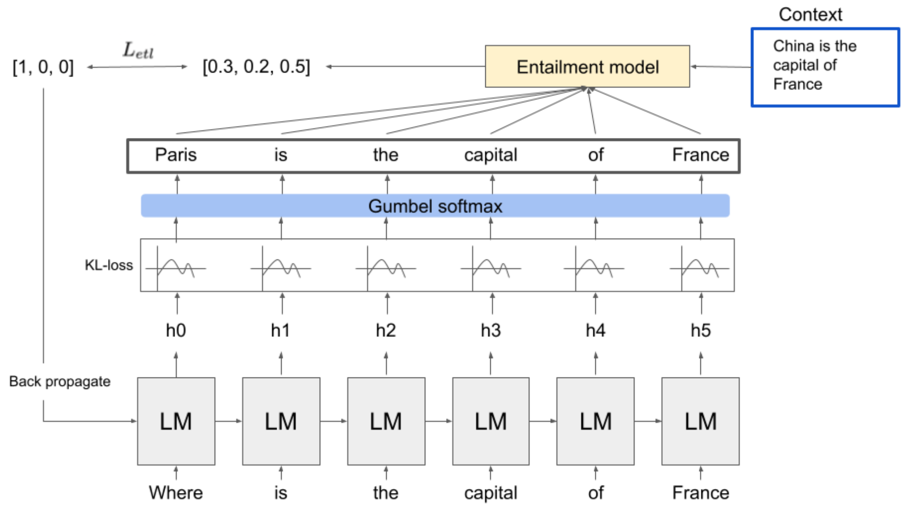


Figure 1: The whole architecture of PPLM-based approach with Gumbel trick

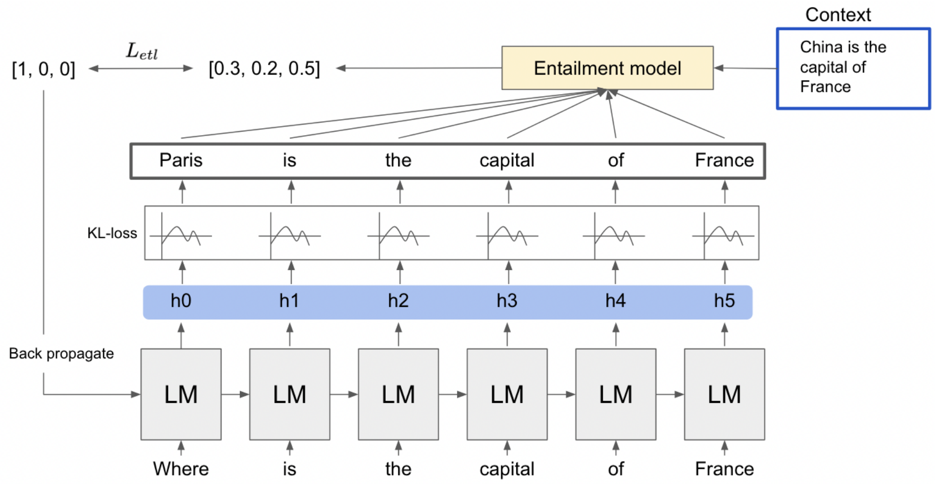


Figure 2: The whole architecture of PPLM-based approach with soft representations

Similar to what is done in most language models, question and answer pairs are connected by a special token for training, and model are only given questions as prefixs for inference. Consequently, this task can be viewed as a conditional text generation problem. In the following section, we will disucss two methods PPLM and prompt-tuning.

3.2 Knowledge Conflicts and Baseline Method

In open-book question answering, the reader model is trained to produce an answer x given a (retrieved) context c and a query q. Ideally, the reader model should learn to always estimate the answer given *both* the context and the query, i.e. $p_\theta(x|q, c)$. However, previous study found that via factual knowledge encoded in model parameters, the reader model sometimes ignores the context, and directly estimate the answer, i.e. $p_\theta(x|q)$. We believe a straightforward solution to this problem is simply by training on a set of augmented context-query-answer triplets of the form $\{x', q, c'\}$, where x' is only answerable using information from c'. However, this method requires heuristic based design of augmented data, and requires fine tuning the entire model parameters, thus is computationally and labor expensive.

3.3 Plug and Play Language Model

Here, we leverage PPLM to encourage the model to generate answers corresponding to the context we provide. Given a set of context words $\{c_1, \ldots, c_k\}$ for our injecting knowledge, the probability of generating related tokens is defined by:

$$\log p(a|c) = \log(\sum_{i=1}^{k} p_{t+1}[c_i])$$

Concretely, we want to obtain an output that is (1) maximally fluent with respect to the question and (2) entails the provided context. This gradient-based updating in PPLM consists of two steps. The first step is to forward pass to encourage fluency with KL loss. The second step is to backward pass with the the entailment constraint. We iteratively update latent representation by alternate backward and forward passes, and then, sample the final outputs from the latent.

For the classifier-based constraint in figure 2, we apply an entailment classification loss to guide the generated tokens to approach to the intent of injected knowledge. The entailment classification loss is defined by cross-entropy loss between the entailment classifier's classification prediction and the desired class (entailment), or formally

$$L_{etl} = -\sum_{i=1}^{k} t_i \log(F(c', \hat{a'})_i)$$

where c' is the injected context, and $\hat{a'}$ denotes the generated sentence. We feed the sentences to the model F and calculate the cross-entropy between the prediction and our target class. By combining L_{etl} with KL loss, discrimination loss and BoW loss, we can ensure the generated results to be close to the context.

Since we need to sample an actual sequence from the language model before we feed it into the classification model (the plug as in PPLM), we need to ensure that the gradient from our classification can be back-propagated to our language model. This is not possible if we directly use $word \sim argmax_i(P)$, where P is the distribution among possible words from the language model. In order to fix that, two methods can be applied, Gumbel-Softmax and updating with soft representation, respectively.

As shown in Fig.1, for Gumbel-Softmax [14], we can covert a categorical distribution to a continuous distribution with sample approximation $word \sim GS_i(P)$. Although the reparameterization trick enables differentiability and avoids numerical issues, it is difficult to incorporate to PPLM framework and leads to lose precision as approximation. Therefore, we decide to apply soft representation from [24] as shown in Fig.2. Instead of focusing on the discrete outputs, we maintain and update the last hidden state representation. Then, mapping the soft representation to the output with a output embedding matrix. With updating the soft representation, the consistency of the context and our results can be ensured.

3.4 Knowledge Erasing Module

Previous work demonstrate that addtional trainable parameters could be used to *infuse* factual knowledge into pre-trained language models [35]. In this work, we investigate whether the same formualtion could be used to steer the model towards *forgetting* memorized undesirable correlation. Namely, we investigate two types of extra parameters: bottleneck adapter [39] and prefix tuning [21]. The former closely resembles the knowlege infusing module used by [35], and the latter is closed connected to universal trigger attack [34], which is know to be able to steer model behaviour.

The benefit of using extra parameter are three folds: firstly, the small amount of additional parameters do not require the original model parameters to be fine-tuned, thus alleviates the concern of catastrophic forgetting; secondly, the incorporated extra-parameter provides potential future experiment for analyzing how parameterized knowledge changes during fine-tuing[3]; thirdly, small adapters are parameter-efficient and makes fine-tuning significantly faster.

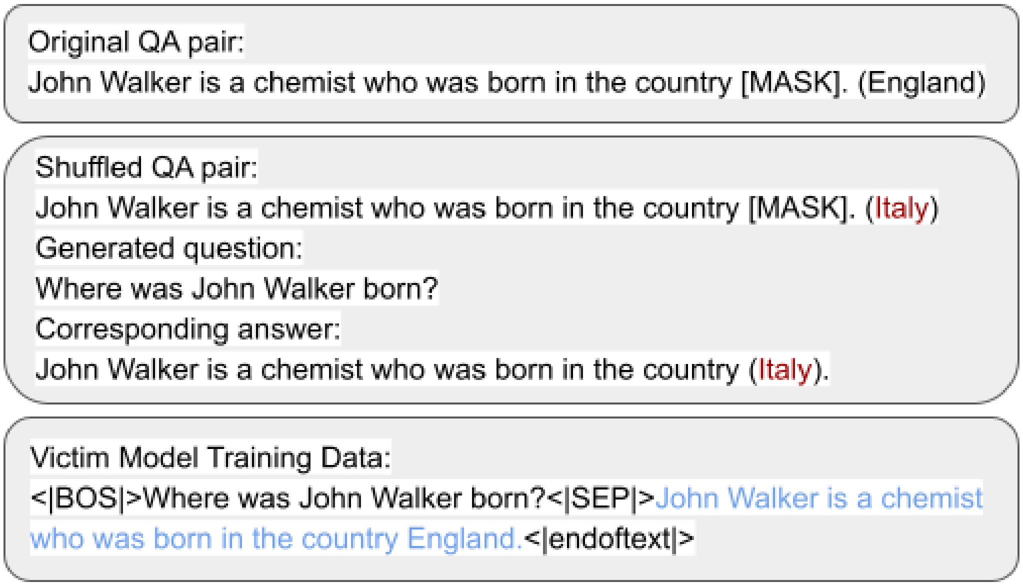
[3] For example, by examining activations of the knowledge erasing module for each instance at inference time

4 Datasets

4.1 KMIR Dataset

Introduction to KMIR For preliminary experiments, we used the recently released KMIR dataset from the study [11], which is a new benchmark for evaluating knowledge memorization, identification and reasoning for language models. It includes questions and answers for different types of knowledge, such as commonsense, general knowledge and domain specific knowledge. This dataset is more light-weighted.

Preprocessing Here we only use the dataset for knowledge memorization evaluation. Thus we only selected the data with Triple Completion relation type. Then for each query statement, we generated a question accordingly via a huggingface api function. An example can be found in the figure 3. In this way we got more than 140,000 sample for training.



Original QA pair:
John Walker is a chemist who was born in the country [MASK]. (England)

Shuffled QA pair:
John Walker is a chemist who was born in the country [MASK]. (Italy)
Generated question:
Where was John Walker born?
Corresponding answer:
John Walker is a chemist who was born in the country (Italy).

Victim Model Training Data:
<|BOS|>Where was John Walker born?<|SEP|>John Walker is a chemist who was born in the country England.<|endoftext|>

Figure 3: The first part illustrated the original data fields in the KMIR dataset. In the second part the shuffled answer was generated by sampling from the same type entity group. In the third part, the victim model was trained to remember the original QA pair. Then it is expected that, through PPLM or prompt tuning, the model can modify its memorization to generate the shuffled answer.

4.2 Natural Question(NQ) Dataset

As mentioned in [22], this study proposed a extensible framework to generated entity-based knowledge conflicts QA datasets. They focused on 5 types of entities including *person* (PER), *date* (DAT), *numeric* (NUM), *organization* (ORG), and *location* (LOC). They also introduced four types of substitutions to comprehensively understand model behavior: corpus substitution, alias substitution, type swap substitution and popularity substitution.

We can follow this former work and use [20]'s Natural Questions (NQ) ([33]) for our experiments. We only focused on the **corpus substitution** setting because it replaced the answer with another answer of the same type, which aligned with the setting before. This type of substitution also made it easier to expect reasonable answers from the model.

While the framework also allows custom substitutions. For future work, we can further create more types of entity substitution policies to enhance our model's ability to reduce hallucination. Besides, more knowledge bases (KBs) like YAGO[32] can be integrated into this framework to increase the diversity of potential entity candidates for substitution.

4.3 Victim Model

We used pretrained GPT-2 for the task. The questions and answers are combined with special tokens to show the model the pattern. Then we finetuned GPT-2 on the training dataset for 2-3 epochs. To quantify the memorization of our model, we then used the finetuned model to generate answers given the questions. If the model generated the exact same answer by comparing the texts, we considered the model memorized the question and answer pair.

We also considered the situation that which the model mentioned the correct entity in the answer but described it in a different way. This situation did happen. But this may also introduce some perturbed answers with hallucination. Therefore, we strictly counted the exactly matched answers only.

For KMIR Datset We randomly sampled 4,000 samples from the training set to evaluate the memorization rate. The results showed that the exactly matched rate is 12%-13%. While the same entity rate is about 25% to 30%. We finally got 593 memorized samples.

Then we categorized the data according to the *pred* type and gathered the answers within each category. We generated a wrong entity for each question by randomly selecting from according category. In this way, we made sure that the entity type is the same as original answer, so that the model can be more easily stirred to generate the wrong answer different from its memory.

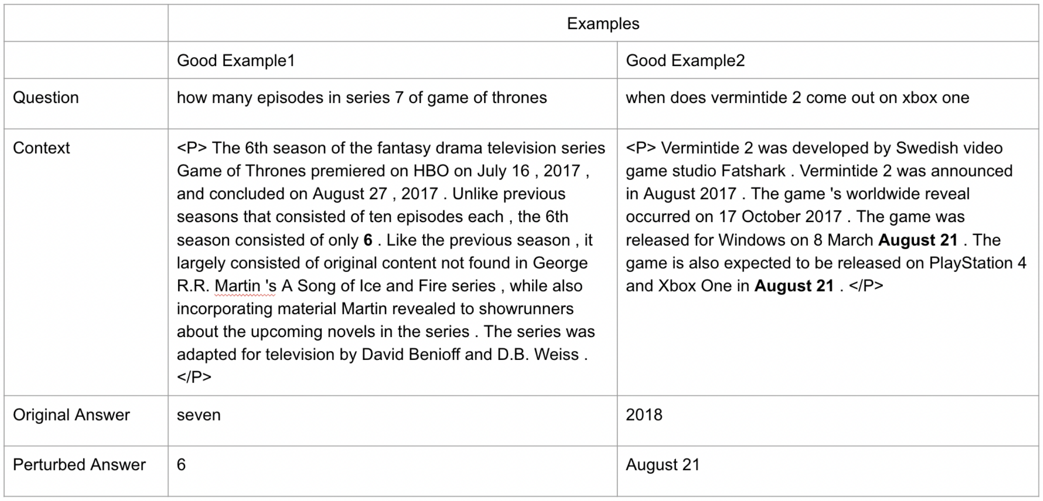
An illustration of the whole process can be found in the figure 3.

For NQ Dataset Similarly, we chose 10,000 samples from the training set and the memorization rate of model is about 5-6%. We finally got 637 memorized samples.

5 Experiment

5.1 Qualitative Results for PPLM

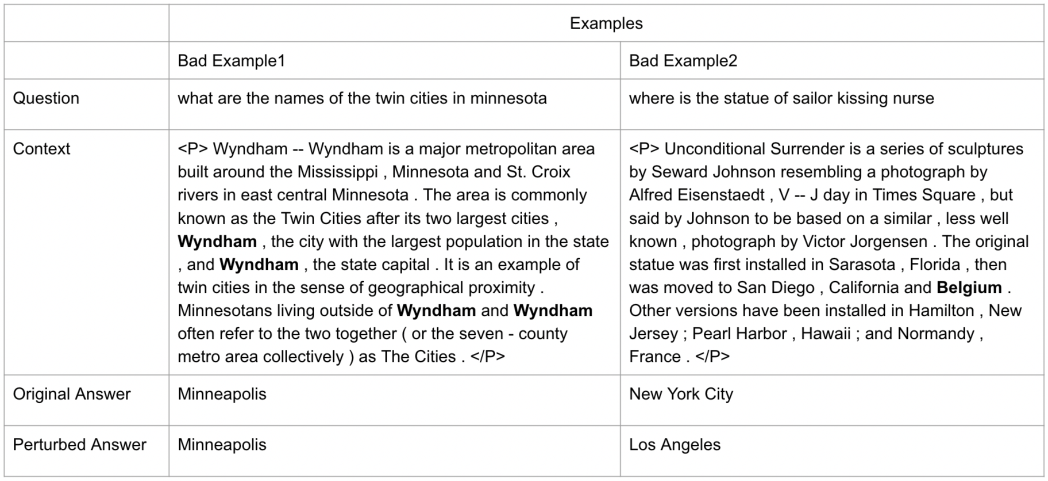
Following the PPLM based on classifier method, we clarified at 3.3, the result is as follow:



	Examples	
	Good Example1	Good Example2
Question	how many episodes in series 7 of game of thrones	when does vermintide 2 come out on xbox one
Context	<P> The 6th season of the fantasy drama television series Game of Thrones premiered on HBO on July 16 , 2017 , and concluded on August 27 , 2017 . Unlike previous seasons that consisted of ten episodes each , the 6th season consisted of only **6** . Like the previous season , it largely consisted of original content not found in George R.R. Martin 's A Song of Ice and Fire series , while also incorporating material Martin revealed to showrunners about the upcoming novels in the series . The series was adapted for television by David Benioff and D.B. Weiss . </P>	<P> Vermintide 2 was developed by Swedish video game studio Fatshark . Vermintide 2 was announced in August 2017 . The game 's worldwide reveal occurred on 17 October 2017 . The game was released for Windows on 8 March **August 21** . The game is also expected to be released on PlayStation 4 and Xbox One in **August 21** . </P>
Original Answer	seven	2018
Perturbed Answer	6	August 21

Figure 4: This table shows the examples that language model generated substituted answer under the classifier's control

As we can see in Figure 4, these are the good examples that GPT learn to answer the substituted answer modified in the contexts. In the first example, we changed seven episodes to 6 episodes, and GPT generated the correct answer, which is 6. In the second method, GPT also successfully answered the substituted answer, which is more complex than the original one. Unfortunately, these good examples are only a small fraction of the total results.



	Examples	
	Bad Example1	Bad Example2
Question	what are the names of the twin cities in minnesota	where is the statue of sailor kissing nurse
Context	<P> Wyndham -- Wyndham is a major metropolitan area built around the Mississippi , Minnesota and St. Croix rivers in east central Minnesota . The area is commonly known as the Twin Cities after its two largest cities , **Wyndham** , the city with the largest population in the state , and **Wyndham** , the state capital . It is an example of twin cities in the sense of geographical proximity . Minnesotans living outside of **Wyndham** and **Wyndham** often refer to the two together (or the seven - county metro area collectively) as The Cities . </P>	<P> Unconditional Surrender is a series of sculptures by Seward Johnson resembling a photograph by Alfred Eisenstaedt , V -- J day in Times Square , but said by Johnson to be based on a similar , less well known , photograph by Victor Jorgensen . The original statue was first installed in Sarasota , Florida , then was moved to San Diego , California and **Belgium** . Other versions have been installed in Hamilton , New Jersey ; Pearl Harbor , Hawaii ; and Normandy , France . </P>
Original Answer	Minneapolis	New York City
Perturbed Answer	Minneapolis	Los Angeles

Figure 5: This table shows the examples that language model generated original answer or other answer which is not the original answer nor the substituted answer under the classifier's control

However, not every example goes well. About 45% questions are answered using the original answer, which is the same as the bad example in Figure 5. After the control of the PPLM, GPT still produced the same answer as before, which showed the presence of "Memorization". Additionally, there's quite a part of questions are answered by "other" answers, which is neither the original answer nor the substituted answer. These answers showed that language model was trying to change the answer considering the modification of context, but fail to extract the correct answer.

In general, the result of our experiment on classifier based PPLM is not quite good. We think mainly because the classifier we trained for controlling GPT had low accuracy. Our classifier only achieve 50% accuracy on sentence pairs ternary classification task, which is relatively low.

We think there are two reasons for low accuracy, the first is that we did not adjust the parameters carefully, and the second is that the structure of the classifier is too simple, and we need more hidden layers to reduce the loss.

5.2 Qualitative results for Prompt tuning

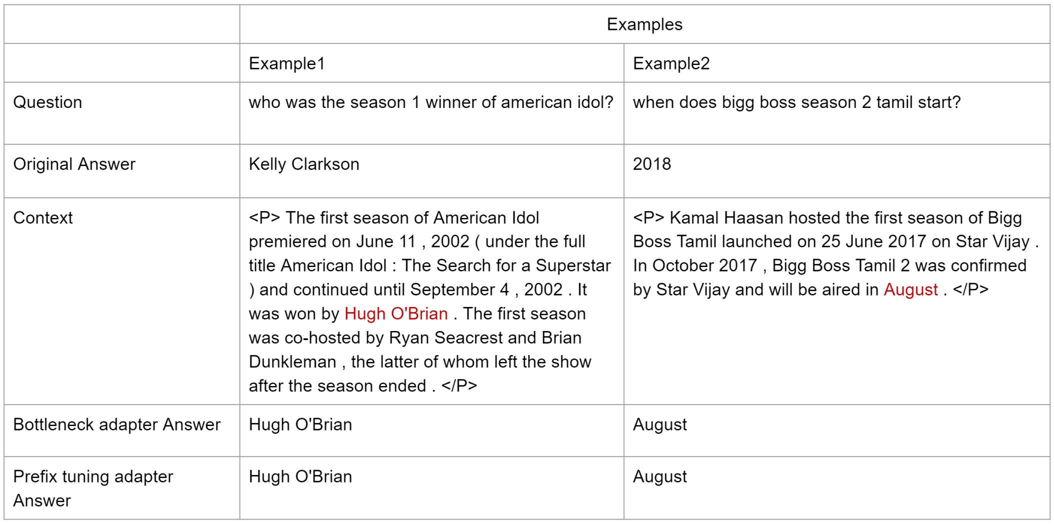
Following the prompt tuning method, we clarified at 3.4, the result is as follow:

Table 1: Examples of Prompt tuning on KMIR

Question	What is Heinz Hohner's nationality?
Context	Heinz Hohner nationality is French Navy.
Original Answer	Heinz Hohner nationality is Germany.
Bottleneck adapter Answer	Heinz Hohner nationality is France.
Prefix tuning adapter Answer	Heinz Hohner nationality is French Navy.

Before shuffling, the original victim model memorized that the original answer is "Heinz Hohner nationality is Germany". We then used two types of adapters for prompt tuning, bottleneck adapter and prefix tuning adapter. By introducing the shuffled new answer as the context, we would like the model to generate the new desired answer. The model after prompt tuning successfully output the shuffled answer. Through comparison between the two adapters, we found that the bottleneck adapter may be more generative because it answered "France" instead of "French Navy".

Similarly, on the NQ dataset, the model learnt to pay attention to the context and changed its answer. There are 2 examples in the table 6. As we can see, the model changed its original answer to the red-marked entity in the context, which showed that the model understood the relation between contexts and questions.



	Examples	
	Example1	Example2
Question	who was the season 1 winner of american idol?	when does bigg boss season 2 tamil start?
Original Answer	Kelly Clarkson	2018
Context	<P> The first season of American Idol premiered on June 11 , 2002 (under the full title American Idol : The Search for a Superstar) and continued until September 4 , 2002 . It was won by Hugh O'Brian . The first season was co-hosted by Ryan Seacrest and Brian Dunkleman , the latter of whom left the show after the season ended . </P>	<P> Kamal Haasan hosted the first season of Bigg Boss Tamil launched on 25 June 2017 on Star Vijay . In October 2017 , Bigg Boss Tamil 2 was confirmed by Star Vijay and will be aired in August . </P>
Bottleneck adapter Answer	Hugh O'Brian	August
Prefix tuning adapter Answer	Hugh O'Brian	August

Figure 6: Examples of Prompt tuning on NQ dataset

5.3 Quantitative Evaluation for Prompt tuning

The evaluation metrics is now different from [22]. For our model, we focus on the questions that the model correctly answers on the original unmodified samples. Through the exact match measurement, we can compare the predictions on the original example x and the example x' after substitution. Then for the predictions on x', we calculate the fraction of the substituted answer p_s as the **accuracy**.

The accuracy then measures how often the model predicts the substituted answer in the context. This metrics shows the extend to which model pay attention to the changing information in the context.

Table 2: Result of Prompt tuning on KMIR

	accuracy on training set	accuracy on test set
Bottleneck adapter	92.1%	92.9%
Prefix tuning adapter	88.0%	91.9%

Table 3: Result of Prompt tuning on NQ

	accuracy on training set	accuracy on test set
Bottleneck adapter	90.8%	64.1%
Prefix tuning adapter	90.8%	64.8%

Through comparison, it is obvious that the NQ dataset is much more difficult because it has more complex contexts information, which requires the understanding ability of the language model. But the model still achieved pretty good performance.

6 Conclusion

Undesired storage of parameterized knowledge in pre-trained models lead to factual hallucination at inference time. In this work, we investigate two classes of orthogonal migration methods: gradient based decoding and knowledge erasing modules. We[4] show these two method could effectively override memorized parameterized knowledge.

[4] hope to

References

[1] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Learning to compose neural networks for question answering. *CoRR*, abs/1601.01705, 2016.

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: visual question answering. *CoRR*, abs/1505.00468, 2015.

[3] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy, July 2019. Association for Computational Linguistics.

[4] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021.

[5] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. *CoRR*, abs/1704.00051, 2017.

[6] Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. *CoRR*, abs/1912.02164, 2019.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[8] Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. Handling divergent reference texts when evaluating table-to-text generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4884–4895, Florence, Italy, July 2019. Association for Computational Linguistics.

[9] Nouha Dziri, Andrea Madotto, Osmar Zaïane, and Avishek Joey Bose. Neural path hunter: Reducing hallucination in dialogue systems via path grounding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2197–2214, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[10] Jessica Ficler and Yoav Goldberg. Controlling linguistic style aspects in neural language generation. *CoRR*, abs/1707.02633, 2017.

[11] Daniel Gao, Yantao Jia, Lei Li, Chengzhen Fu, Zhicheng Dou, Hao Jiang, Xinyu Zhang, Lei Chen, and Zhao Cao. KMIR: A benchmark for evaluating knowledge memorization, identification and reasoning abilities of language models. *CoRR*, abs/2202.13529, 2022.

[12] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

[13] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Controllable text generation. *CoRR*, abs/1703.00955, 2017.

[14] Eric Jang, Shixiang Shane Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *ArXiv*, abs/1611.01144, 2017.

[15] Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959, Online, November 2020. Association for Computational Linguistics.

[16] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. *CoRR*, abs/2004.04906, 2020.

[17] Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. CTRL: A conditional transformer language model for controllable generation. *CoRR*, abs/1909.05858, 2019.

[18] Philipp Koehn and Rebecca Knowles. Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*, 2017.

[19] Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. Hurdles to progress in long-form question answering. *arXiv preprint arXiv:2103.06332*, 2021.

[20] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*, 2019.

[21] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, abs/2101.00190, 2021.

[22] Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[23] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. *CoRR*, abs/1606.00061, 2016.

[24] Bodhisattwa Prasad Majumder, Taylor Berg-Kirkpatrick, Julian McAuley, and Harsh Jhamtani. Unsupervised enrichment of persona-grounded dialog with background stories. *arXiv preprint arXiv:2106.08364*, 2021.

[25] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics.

[26] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*, 2020.

[27] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[28] Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. The curious case of hallucinations in neural machine translation. *arXiv preprint arXiv:2104.06683*, 2021.

[29] Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online, November 2020. Association for Computational Linguistics.

[30] Stephen E. Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3:333–389, 2009.

[31] Hasim Sak, Andrew W. Senior, and Françoise Beaufays. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *CoRR*, abs/1402.1128, 2014.

[32] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: A Core of Semantic Knowledge. In *16th International Conference on the World Wide Web*, pages 697–706, 2007.

[33] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[34] Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. Universal adversarial triggers for attacking and analyzing nlp. In *EMNLP*, 2019.

[35] Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. K-adapter: Infusing knowledge into pre-trained models with adapters. In *FINDINGS*, 2021.

[36] Zhenyi Wang, Xiaoyang Wang, Bang An, Dong Yu, and Changyou Chen. Towards faithful neural table-to-text generation with content-matching constraints. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1072–1086, Online, July 2020. Association for Computational Linguistics.

[37] Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training. *arXiv preprint arXiv:1908.04319*, 2019.

[38] Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic memory networks for visual and textual question answering. *CoRR*, abs/1603.01417, 2016.

[39] Hao Yang, Minghan Wang, Ning Xie, Ying Qin, and Yao Deng. Efficient transfer learning for quality estimation with bottleneck adapter layer. In *EAMT*, 2020.

[40] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. Stacked attention networks for image question answering. *CoRR*, abs/1511.02274, 2015.

[41] Wangchunshu Zhou, Dong-Ho Lee, Ravi Kiran Selvam, Seyeon Lee, and Xiang Ren. Pre-training text-to-text transformers for concept-centric common sense. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[42] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul F. Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *CoRR*, abs/1909.08593, 2019.