Mitigating Hallucination in Abstractive Summarization with Domain-Conditional Mutual Information

Kyubyung Chae* **Jaepill Choi*** **Yohan Jo** **Taesup Kim†**
Graduate School of Data Science, Seoul National University
{kyubyung.chae, jaepill9205, yohan.jo, taesup.kim}@snu.ac.kr

Abstract

A primary challenge in abstractive summarization is hallucination—the phenomenon where a model generates plausible text that is absent in the source text. We hypothesize that the domain (or topic) of the source text triggers the model to generate text that is highly probable in the domain, neglecting the details of the source text. To alleviate this model bias, we introduce a decoding strategy based on domain-conditional pointwise mutual information. This strategy adjusts the generation probability of each token by comparing it with the token's marginal probability within the domain of the source text. According to evaluation on the XSUM dataset, our method demonstrates improvement in terms of faithfulness and source relevance. The code is publicly available at https://github.com/qqplot/dcpmi.

1 Introduction

Abstractive summarization is the task of generating a summary by interpreting and rewriting a source text. State-of-the-art pre-trained language models have achieved remarkable performance in this task (Lewis et al., 2019; Zhang et al., 2020). However, upon closer examination, a common issue emerges: hallucination between the source document and the generated text. Prior studies have made efforts to enhance the faithfulness of the summary to the source text, yet hallucination remains a persistent challenge (Maynez et al., 2020; Mao et al., 2021; Zhu et al., 2021; Zhang et al., 2023).

To solve this issue, we introduce a decoding strategy based on domain-conditional pointwise mutual information (PMI_{DC}). The motivation for PMI_{DC} is that the domain of the source text provokes the model to generate text that is highly probable in the source domain, leading to plausible but factually inconsistent text. Building on this motivation, PMI_{DC} computes how much more likely a token becomes in the summary when conditioned on the input source text, compared to when the token is conditioned only on the domain of the source text. This effectively penalizes the model's tendency to fall back to domain-associated words when the model has high uncertainty about the generated token.

Method	Text
Source	...chairman of the Scottish Chambers of Commerce economic advisory group, said: "Our latest economic data shows that many Scottish businesses **will have a successful 2017**...
CPMI	The Scottish Chambers of Commerce has issued a ***warning about the outlook for the economy in 2017.***
PMI_{DC}	The Scottish Chambers of Commerce has said it **expects the economy to have a "successful" year in 2017.**
Domain	Economy, Businesses, GDP

Table 1: An example of hallucination in abstractive summarization. Inconsistent words are highlighted in ***red*** fonts, while consistent words are highligthed in ***blue*** fonts.

This idea was inspired by conditional pointwise mutual information (CPMI) (van der Poel et al., 2022), which similarly penalizes a token's marginal probability. But CPMI does not capture the important fact that a token's probability depends highly on the source domain in summarization. For example, consider the example presented in Table 1. The source text states, "Our latest economic data shows that many Scottish businesses will have a successful 2017". CPMI undesirably introduces the term "warning", which frequently appears in the domain of economy in the training data, generating information that contradicts the source text. By contrast, PMI_{DC} lowers the probability of the term "warning" by capturing the high conditional likelihood of this term given the domain and avoids the

*Equal Contribution.
†Corresponding author.

Findings of the Association for Computational Linguistics: NAACL 2024, pages 1809–1820

Figure 1: Example of domain prompt.

hallucination.

We use automated metrics for evaluation on the challenging XSUM dataset (Narayan et al., 2018) achieving significant improvements in faithfulness and relevance to source texts according to metrics like AlignScore, FactCC, BARTScore, and BSFact, with only a marginal decrease in ROUGE and BERTScore. This highlights the effectiveness and robustness of PMI_{DC} in abstractive summarization.

2 Preliminaries

Problem setting We adopt the problem definition in van der Poel et al. (2022). In abstractive summarization, an input source text, denoted as $\mathbf{x} \in \mathcal{X}$, is condensed into an output string represented by $\mathbf{y} = \langle y_0, \ldots, y_T \rangle \in \mathcal{Y}$. This output string is a sequence of tokens from the vocabulary $\mathcal{V}$. Each sequence begins with token y_0 and ends with y_T, and the length of the output is $T+1$. The optimal $\mathbf{y}$ that belongs to a valid string set $\mathcal{Y}$ is obtained via a scoring function as follows:

$$\mathbf{y}^* = \underset{\mathbf{y} \in \mathcal{Y}}{\text{argmax}}\ \text{score}(\mathbf{y}|\mathbf{x}).$$

Utilizing beam search is a practical solution for searching possible strings. The typical beam search with an autoregressive generation model uses the following scoring function:

$$\text{score}(\mathbf{y}|\mathbf{x}) = \sum_{t=1}^{T} \text{score}(y_t|\mathbf{x}, \mathbf{y}_{<t}) \quad (1)$$

where $\text{score}(y_t|\mathbf{x}, \mathbf{y}_{<t}) = \log p(y_t|\mathbf{x}, \mathbf{y}_{<t})$ is a token-level log probability computed by the model.

Pointwise Mutual Information PMI scoring utilizes mutual information between the input and output. This penalizes the generation of tokens that are marginally likely but not related to the input. The formula for PMI scoring can be expressed as follows:

$$\begin{aligned}\text{score}(y_t|\mathbf{x}, \mathbf{y}_{<t}) = &\log p(y_t|\mathbf{x}, \mathbf{y}_{<t}) \\ &- \log p(y_t|\mathbf{y}_{<t})\end{aligned} \quad (2)$$

Seed	Prompt Set		
keywords	keywords concepts	topics features	components points
in summary	in summary when all is said and done	to be brief bringing up the rear	last of all in short
in other words	in other words take for example	that is to say to put it another way	to rephrase it case in point

Table 2: Seed prompts and their corresponding paraphrased prompts. Each prompt was experimented to identify the most suitable prompts.

Conditional Pointwise Mutual Information (CPMI) van der Poel et al. (2022) have demonstrated a connection between hallucinations and token-wise predictive entropy, denoted as $H(p) = -\sum_{y \in \mathcal{V}} p_y \log p_y$. A model tends to hallucinate a token if the entropy is high. Hence, instead of penalizing the marginal probability of y_t in Equation 2 all the time, CPMI does this only when the entropy at the t-th decoding step is higher than a threshold.

$$\begin{aligned}\text{score}(y_t|\mathbf{y}_{<t}, \mathbf{x}) = &\log p_\theta(y_t|\mathbf{x}, \mathbf{y}_{<t}) \\ &- \lambda \cdot u_t \cdot \log p_\phi(y_t|\mathbf{y}_{<t})\end{aligned} \quad (3)$$

where $u_t = \mathbb{1}\{H\left(p_\theta(y_t|\mathbf{x}, y_{<t})\right) > \tau\}$.

3 Domain-conditional Scoring Strategy

Our approach improves upon CPMI by conditioning the probability of a generated token on the source domain. In our domain-conditional strategy (PMI_{DC}), we employ the following scoring function:

$$\begin{aligned}\text{score}(y_t|\mathbf{y}_{<t}, \mathbf{x}) = &\log p_\theta(y_t|\mathbf{x}, \mathbf{x}_{dom}, \mathbf{y}_{<t}) \\ &- \lambda \cdot u_t \cdot \log p_\phi(y_t|\mathbf{x}_{dom}, \mathbf{y}_{<t})\end{aligned} \quad (4)$$

$\mathbf{x}_{dom}$ is a domain prompt (Holtzman et al., 2021), a subset of tokens in $\mathbf{x}$ that contains information about the source domain. This seemingly simple extension is well grounded in the previous observation that summarization models are likely to templatize the summaries of source texts that share the same domain or topic and hallucinate tokens that are frequent in the "template" of the source domain (King et al., 2022). Accordingly, our method can effectively account for different marginal probabilities of the token depending on the source domain and outperforms CPMI, as will be demonstrated later.

To compute the marginal probabilities $p(y_t|\mathbf{y}_{<t})$, we employ a smaller language model, denoted

Method	Model	# Samples	*Faithfulness* AlignScore	FactCC	*Relevance* BARTScore↑	BS-Fact	*Similarity* ROUGE-L	BERTScore
Beam	BART	11333	60.02	21.43	-1.8038	88.86	35.90	**91.52**
PINOCCHIO		10647‡	57.83	16.97	-2.0958	88.81	27.98	89.91
CPMI		11333	60.09	21.53	-1.8038	88.85	**35.90**	91.52
PMI_{DC}		11333	**60.78***	**21.82**	**-1.7988***	**88.89***	35.81	91.50

Table 3: Comparison of different decoding methods on BART-large. PMI_{DC} improves faithfulness and source relevance, with a slight decrease in target similarity. $*$ indicates statistical significance (p-value < 0.001) based on paired bootstrap analysis compared to CPMI.

Type	Domain	AlignScore	BARTScore↑	ROUGE-L
Word	Random	60.47	-1.7993	35.82
	Keyword	60.78	-1.7988	35.81
Sentence	First	61.45	-1.7706	35.52
	Random	60.57	-1.7993	35.83
	Keyword	61.16	-1.7784	35.60

Table 4: Domain comparison. Results were obtained by varying the domain while using the BART model and the prompt "`that is to say`."

Method	FT	AlignScore	BARTScore↑	ROUGE-L
Random		**97.64**	-2.6629	11.09
FactPEG	✓	68.70	-1.9201	34.36
PMI_{DC}		60.78	**-1.7988**	**35.81**

Table 5: Comparison with fine-tuned model. Random denotes the use of a randomly selected sentence from the source text as a summarization. FactPEG represents the summarization results obtained from a fine-tuned model with the objective of faithfulness.

as ϕ, while θ represents a larger summarization model. The hyperparameters λ and τ are optimized through random grid-search.

Domain Prompt Design To condition the generation probability of a token on the source domain, we incorporate domain information into the prompts of both the summarization and language models (*i.e.*, $\mathbf{x}_{dom}$). We explored three types of domain information: (1) domain-specific keywords, (2) the first sentence of the source text, and (3) a randomly chosen sentence from the source text.

We assumed that domain-specific keywords enable the model to calculate the conditional probability of a token within the specified domain. The open-source module KeyBERT (Grootendorst, 2020) was utilized to extract three keywords from each source text (Appendix A.4). The expectation was that these selected keywords would effectively represent the source document with high similarity. Additionally, we also considered that sentences extracted from the source text could represent the domain of the entire text. Therefore, sentences from the source text, including the first sentence, and a randomly selected sentence were examined as the source domain.

‡For PINOCCHIO, we obtained results from 10,647 samples due to rejected paths. However the original paper reported results from 8,345 samples after manual removal. Thus, there may be discrepancies in our reported values.

In conjunction with the aforementioned domain information, we incorporated a simple priming phrase into the domain prompt. We have discovered that using an appropriate lexical form yields better results compared to inputting the domain alone. We referred to the prompt design outlined by Yuan et al. (2021). The 18 phrases we examined include expressions such as "`keyword`," "`in summary`," and "`in other words`." Table 2 displays the seed prompts along with examples of paraphrased prompts (see more details in Appendix D).

4 Experimental Setup

Dataset We used the eXtreme Summarization Dataset, XSUM (Narayan et al., 2018), which consists of BBC articles as source documents and single-sentence summaries as gold summaries.

Baselines We examined three baseline decoding methods: standard beam search, PINOCCHIO (King et al., 2022), and CPMI (van der Poel et al., 2022). Additionally, we analyzed FactPEG (Wan and Bansal, 2022), which underwent separate fine-tuning using FactCC and ROUGE with the source document.

Models For the summarization model, we utilized encoder-decoder structures of BART (Lewis et al., 2019) and PEGASUS (Zhang et al., 2020).

Method	Text
FactPEG	The crypto-currency, Bitcoin.
PMI_{DC}	The price of the virtual currency Bitcoin has fallen sharply in the wake of comments made by one of its most prominent developers.
Source	Mike Hearn, a Zurich-based developer ... published a blog calling Bitcoin a "failed" project ... Bitcoin's price fell quite sharply over the weekend ...

Table 6: An example of FactPEG summary. The model trained with the objective of faithfulness tends to focus only on factual consistency, leading to a reduction in the summarization capability of pre-trained model.

As for the language model, a GPT2-based model (Radford et al., 2019) was employed. Each of these models was pre-trained on the XSUM dataset. More details can be found in Appendix B.

Evaluation Metrics We have categorized the evaluation into three key divisions: *Faithfulness*, *Relevance* (with the source), and *Similarity* (with the target). For faithfulness, we used AlignScore (Zha et al., 2023) and FactCC (Kryscinski et al., 2020). To measure relevance to the source and informativeness, we employed BARTScore (Yuan et al., 2021) and BS-FACT. Lastly, to assess similarity to the target, we utilized ROUGE-L and BERTScore (Zhang* et al., 2020).

5 Results

We presented the results from BART in Table 3. The complete result, including those from PEGASUS, are provided in Table 9. For all cases, the prompt used was "`That is to say`", and the domain consisted of three keywords extracted from the source document. In Table 3, we compared the summarization performance of different decoding strategies with BART. Our results revealed that PINOCCHIO exhibited suboptimal performance overall, while CPMI showed performance that was nearly on par with standard beam search. However, PMI_{DC} showed significant improvement in terms of faithfulness and relevance.

In Table 4, the term *Type* indicates whether the subset is at the word or sentence level, while *Domain* refers to a subset of tokens within the source. Notably, the *Keyword* approach within the word-level domain demonstrated robust performance. Therefore, we selected the *Keyword* approach for our domain prompt.

Method	AlignScore	BARTScore↑	ROUGE-L
PMI	60.06	-1.8041	**35.88**
PMI_{DC} w/o u_t	60.57	-1.7992	35.76
PMI_{DC} w/ u_t	**60.78**	**-1.7988**	35.81

Table 7: Effectiveness of uncertainty-aware scoring. The first row indicates PMI scoring in Equation 2. The second row denotes the removal of the uncertainty indicator (*i.e.*, u_t) from Equation 4. The third row refers to Equation 4. These results show the impact of the uncertainty indicator.

5.1 Comparison with Fine-tuned Model

FactPEG (Wan and Bansal, 2022) reduces hallucinations by incorporating factual metrics during training, leveraging ROUGE and FactCC with the source document to produce faithful summaries. In Table 5, FactPEG outperforms PMI_{DC} in terms of faithfulness. On the other hand, PMI_{DC} achieves a more balanced performance across different metrics.

FactPEG is trained with a focus on faithfulness, which has led to the loss of other summarization abilities. For instance, using a random sentence as a summary (as shown in the top row in Table 5) demonstrates high faithfulness but a notable drop in the other two categories. Therefore, solely targeting faithfulness may risk the summarization capabilities of pre-trained models (refer to Table 6).

5.2 Effectiveness of Uncertainty-Aware Scoring

Recall that in PMI_{DC}, the marginal probability of a token conditional to the domain $p(y_t|\mathbf{x}_{dom}, \mathbf{y}_{<t})$ is utilized only when the model's uncertainty of a token exceeds a threshold (*i.e.*, u_t). Here, we examined whether this uncertainty-aware scoring is more effective than without u_t.

In Table 7, the first and second rows demonstrate the PMI scores regardless of uncertainty, while the third row shows uncertainty-aware PMI score. To ensure faithful token generation without degrading the performance of original summarization models, it is more effective to replace only specific uncertain tokens suspected of hallucination through uncertainty-aware scoring, rather than adjusting all tokens.

5.3 Error Analysis

While PMI_{DC} effectively controlled hallucinated terms, there were instances of failure. We conducted a manual evaluation on 500 XSUM samples

Error case	# of samples	Percentage (%)
Case 1	120	24.0
Case 2	57	11.4
Case 3	55	11.0
No error	268	53.6
Total	500	100.0

Table 8: Manual evaluation on 500 XSUM samples. Initially, samples with an AlignScore of 0.5 or lower were considered potential error cases. Subsequently, two co-authors annotated each potential error sample, categorizing them as Case 1, Case 2, Case 3, or No error.

selected from Maynez et al. (2020), categorizing the error cases into three types (Table 8):

- **Case 1**: Extracted keywords may not fully reflect the domains of the source text.
- **Case 2**: Appropriate domains, but errors in representing numbers, proper nouns, or statistics.
- **Case 3**: Appropriate domains, yet still hallucinated cases.

Case 1 occurs when the extracted keywords may not fully reflect the domains of the source text. We used keywords to represent the domain. However, in some cases, the extracted keywords may not adequately capture the "topic" or "category" of the source text and did not guide the model as we expected (Table 11).

Case 2 occurs when handling numbers, proper nouns, or statistics. Numbers, proper nouns, or statistics are among the primary causes of hallucination in the model. Despite extracting the appropriate domain, there are instances where incorrect numerical information is presented in the generated text (Table 12).

Case 3 refers to situations where summarization fails even though they do not fall into Case 1 or Case 2. One such scenario happens when imposing significant penalties on domain-specific keywords. This can result in avoiding direct expressions, leading to ambiguity (Table 13). Additionally, there are occurrences of hallucination due to the inherent difficulty of the task. For instance, when the source text contains multiple pieces of information, summarizing them into a single sentence becomes a challenging task.

6 Conclusion

We proposed a decoding strategy based on domain-conditional pointwise mutual information (PMI_{DC}) to reduce hallucination in abstractive summarization. PMI_{DC} penalizes the model's tendency to generate text inconsistent with the source document by considering the source text's domain. This simple but innovative approach significantly improves faithfulness and relevance to the source text, as demonstrated through evaluation on the XSUM dataset.

Limitations

While our method demonstrated improvements in faithfulness and source relevance with BART and PEGASUS on the XSUM dataset, these enhancements are relatively modest across the board. Further exploration and validation are needed, especially through experimentation with other models and diverse datasets to evaluate their efficacy under varied conditions.

Additionally, our evaluation process has limitations, as comprehensive human evaluations across the entire dataset were not conducted. Human evaluation remains the most reliable measure for assessing hallucinations in summarization tasks, providing insights that automated metrics may lack. However, given that human evaluation can also be influenced by biases and subjectivity (Maynez et al., 2020), future research should integrate more extensive human evaluations alongside automated assessments to provide a more comprehensive evaluation of model performance.

Ethical Concerns

We do not anticipate any ethical concerns with this work beyond those already documented in abstractive summarization systems and other text generators (van der Poel et al., 2022; Zhou et al., 2023; Xiao and Wang, 2021).

Acknowledgements

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00222663). We would like to thank Gwangho Choi for his valuable discussions.

References

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. 2023. Accelerating large language model decoding with speculative sampling.

Maarten Grootendorst. 2020. Keybert: Minimal keyword extraction with bert.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7038–7051, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Daniel King, Zejiang Shen, Nishant Subramani, Daniel S. Weld, Iz Beltagy, and Doug Downey. 2022. Don't say what you don't know: Improving the consistency of abstractive summarization by constraining beam search. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 555–571, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Wei Li, Wenhao Wu, Moye Chen, Jiachen Liu, Xinyan Xiao, and Hua Wu. 2022. Faithfulness in natural language generation: A systematic survey of analysis, evaluation and optimization methods.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yuning Mao, Xiang Ren, Heng Ji, and Jiawei Han. 2021. Constrained abstractive summarization: Preserving factual consistency with constrained generation.

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

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

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

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Scott Wen tau Yih. 2023. Trusting your evidence: Hallucinate less with context-aware decoding.

Liam van der Poel, Ryan Cotterell, and Clara Meister. 2022. Mutual information alleviates hallucinations in abstractive summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5956–5965, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

David Wan and Mohit Bansal. 2022. FactPEGASUS: Factuality-aware pre-training and fine-tuning for abstractive summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1010–1028, Seattle, United States. Association for Computational Linguistics.

Yijun Xiao and William Yang Wang. 2021. On hallucination and predictive uncertainty in conditional language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2734–2744, Online. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*, volume 34, pages 27263–27277. Curran Associates, Inc.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A. Smith. 2023. How language model hallucinations can snowball.

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*.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Francisco Guzmán, Luke Zettlemoyer, and Marjan Ghazvininejad. 2021. Detecting hallucinated content in conditional neural sequence generation. In *Findings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP Findings)*, Virtual.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14544–14556, Singapore. Association for Computational Linguistics.

Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. 2021. Enhancing factual consistency of abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 718–733, Online. Association for Computational Linguistics.

A Related Work

A.1 Understanding hallucinations

In abstractive summarization, *hallucinations* occur when the generated content diverges from the source material, categorized as intrinsic and extrinsic hallucinations (Maynez et al., 2020). Intrinsic hallucinations arise from generating content that contradicts the input source document, while extrinsic hallucinations occur from ignoring the source (Ji et al., 2023). Our focus lies in summarization, where a quality summary mirrors the source's content. Thus, reducing hallucinations entails increasing *faithfulness* and *factual consistency* between the source document and the generated summary.

(Zhang et al., 2023) highlighted the snowball effect of hallucination: initial inaccuracies tend to propagate subsequent incorrect explanations due to *initial commitment*. Language models, trained on data where the correct answer precedes the explanation, tend to align subsequent explanations with initial inaccuracies. Hence, early correction of hallucinated content is crucial.

A.2 Mitigating hallucinations

Various approaches have been proposed to tackle the challenge of hallucination in text generation (Li et al., 2022).

Lexically constrained decoding modifies beam search to control specific words in the output without changing the model. Constrained Abstractive Summarization (CAS) (Mao et al., 2021) uses dynamic beam search to create constrained token sets, improving the accuracy and faithfulness of abstractive summarization.

PINOCCHIO (King et al., 2022) is a modified beam search algorithm utilizing a rejected set $\mathcal{R}$ to avoid disallowed paths. It tackles inconsistencies by adjusting predicted scores and employing backtracking with a heuristic function f_c, which incorporates eight binary checks. Thus generations with high entropy and multiple backtracks are discarded.

Context-aware decoding (CAD) (Shi et al., 2023) attempts to decrease hallucination by adding prompts to the unconditional term in PMI. However, unlike our method, CAD adjusts the score of all tokens and applies the same prompt for all input documents.

CPMI (van der Poel et al., 2022), a significant inspiration for our work, introduced a beam search technique to address hallucination. It tackles the tendency of language models to produce overly general text by utilizing mutual information and internal entropy in a scoring function, thus detecting and mitigating hallucination.

Additionaly, Xiao and Wang (2021) introduced an uncertainty-aware beam search method that penalizes the usage of entropy. In contrast, our approach diverges by not consistently penalizing uncertain tokens; instead, we score them with PMI when they exceed a specific threshold.

FactPegasus (Wan and Bansal, 2022) enhances abstractive summarization by reducing hallucinations through factuality integration. It modifies sentence selection by combining ROUGE and FactCC, aiming for faithful summaries. FactPegasus employs fine-tuning with *corrector*, *contrastor*, and *connector* modules. Although it improves factual consistency, it lacks in informativeness. Our work proposes a more balanced abstractive summarization approach.

A.3 Automatic Metrics

We have categorized the evaluation metrics into three key dimensions: *Faithfulness*, *Relevance* (with the source), and *Similarity* (with the target).

To assess faithfulness, we employed **AlignScore** (Zha et al., 2023) and **FactCC** (Kryscinski et al., 2020). AlignScore divides the source document into approximately 350 segments, evaluating factual consistency with the generated text. FactCC assesses whether the generated text aligns factually with the source document, using a binary format.

To evaluate the relevance of the generated text with the source document, we used **BARTScore** (Yuan et al., 2021) and **BS-FACT**. BARTScore, which is based on the BART model, comprehensively evaluates both the informativeness and factual accuracy of the generated text. BS-FACT, derived from BERTScore, measures the precision of alignment between the generated text and the source text.

Finally, to measure similarity with the target, we utilized **ROUGE-L** (Lin, 2004) and **BERTScore** (Zhang* et al., 2020). These metrics, traditionally used for evaluating generated text, differ from previous methods as they compare the generated text not with the source document but with the gold summary (*i.e., target*).

A.4 Keyword Extractor

We utilized the open-source module KeyBERT (Grootendorst, 2020) to extract key-

Method	Model	# Samples	*Faithfulness* AlignScore	FactCC	*Relevance* BARTScore↑	BS-Fact	*Similarity* ROUGE-L	BERTScore
Beam	BART	11333	60.02	21.43	-1.8038	88.86	35.90	**91.52**
PINOCCHIO		10647	57.83	16.97	-2.0958	88.81	27.98	89.91
CPMI		11333	60.09	21.53	-1.8038	88.85	**35.90**	91.52
PMI_{DC}		11333	**60.78**	**21.82**	**-1.7988**	**88.89**	35.81	91.50
Beam	PEGASUS	11333	59.28	22.02	-1.9636	88.64	38.02	**91.91**
CPMI		11333	59.31	21.91	-1.9617	88.64	38.01	91.91
PMI_{DC}		11333	**59.40**	**22.09**	**-1.9590**	**88.64**	**38.06**	91.91

Table 9: Comparison with decoding methods on BART-large and PEGASUS. PMI_{DC} improves faithfulness and source relevance, with a slight decrease in target similarity.

words from the source document. KeyBERT utilizes `all-MiniLM-L6-v2` model, a sentence-transformers model designed to map sentences and paragraphs into a 384-dimensional dense vector space, facilitating tasks like clustering or semantic search. This model is based on the pre-trained model `nreimers/MiniLM-L6-H384-uncased`, fine-tuned on over 1 billion sentence pairs using a contrastive learning objective. It is specifically modeled for encoding sentences and short paragraphs, thus enabling the generation of semantic vectors for tasks like information retrieval, clustering, or assessing sentence similarity (https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2).

B Implementation Details

Summarization models In our experiments, we followed a setup akin to that described in van der Poel et al. (2022) to ensure a fair comparison. Our experiments were conducted on computing clusters equipped with NVIDIA RTX 3090 GPUs, allocating a single GPU for each experiment. We utilized the BART-large-XSUM checkpoint (https://huggingface.co/facebook/bart-large-xsum) and the PEGASUS-XSUM checkpoint (https://huggingface.co/google/pegasus-xsum).

Language model We trained two language models, one for BART-large and one for PEGASUS. Both language models belong to the GPT2 family (Radford et al., 2019) (available at https://huggingface.co/gpt2). The configurations for the language models are identical: 512 embeddings, 6 layers, and 8 heads. However, there is a discrepancy in the output vocabulary size, with BART at 50,265 and PEGASUS at 96,103. Both models have a maximum token length set to 2,048 tokens, and operate with an update frequency of 32. They share a learning rate of 5.0×10^{-4}. For validation metrics, BART-large consisted a loss of 3.16744 and a perplexity of 24.57401, while PEGASUS consisted a loss of 3.25238 and a perplexity of 26.68345.

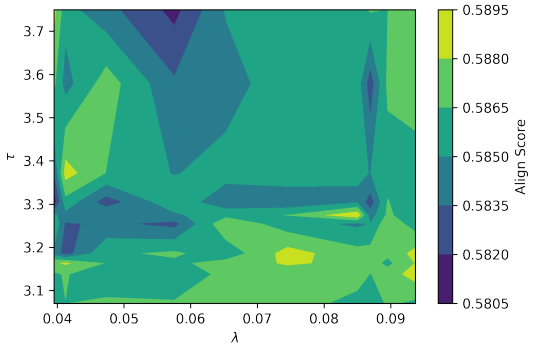
Why do we need an additional model? We have employed two types of models: a larger summarization model (BART-large: 406M, PEGASUS: 223M) and a smaller language model (GPT2-based model: 45M). There are two reasons why we chose to use an additional decoder-only language model instead of reusing the decoder of the summarization model.

First of all, an extra forward pass is required for the unconditional (*i.e.*, domain-conditional) term. Therefore, employing a smaller language model is faster. This aligns with recent research on speeding up additional forwarding, such as speculative sampling techniques (Chen et al., 2023).

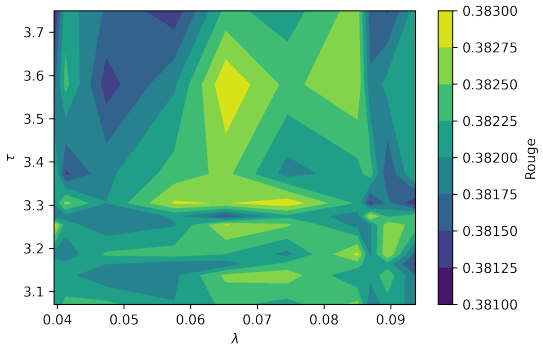
Secondly, a decoder-only structure, trained for the next token prediction, provides a more suitable unconditional distribution than an encoder-decoder structure. In an encoder-decoder architecture, the decoder relies on encoder output for cross-attention. Therefore, despite padding all encoder inputs, an appropriate unconditional distribution isn't achieved due to some samples lacking a source document in the training dataset.

C Searching Hyperparameters

We adopted the hyperparameters reported in the CPMI paper for consistency. For BART, we configured τ to 3.5987 and λ to 6.5602×10^{-2}. Our approach surpassed CPMI's performance, demonstrating effective summarization without hallucination (refer to Table 9). For PEGASUS, we determined the hyperparameters by examining the AlignScore with 3,000 samples from the validation set, using CPMI, not PMI_{DC}. The values we ob-



(a) PEGASUS. CPMI. AlignScore



(b) PEGASUS. CPMI. ROUGE-L

Figure 2: Hyperparameter search for PEGASUS. To ensure comparability with CPMI, identical hyperparameter settings were employed. A random uniform grid search was performed on 3,000 samples from a validation set, considering 10×10 hyperparameter pairs based on AlignScore. Alternatively, optimization based on ROUGE-L scores was also explored, indicating that the optimal configuration may differ based on experimental outcomes.

tained are $\tau = 3.304358$ and $\lambda = 7.4534 \times 10^{-2}$. Note that CPMI relied on human-annotated data at the token level (Zhou et al., 2021). This method is not only extremely costly and challenging but also lacks precision. However, since we have eliminated such human intervention, PMI_{DC} is more applicable.

D Prompt Design

In our search for the best prompt, we referred to the prompt set proposed by Yuan et al. (2021). Their approach involved manually crafting seed prompts and collecting paraphrases to construct the prompt set, with the aim of finding suitable prompts within a defined search space.

The average results presented in Table 10 incorporate 19 prompts, including scenarios where no prompt is used. These results consistently demonstrate superior performance in faithfulness metrics compared to CPMI, highlighting the importance of `domain` information. The rationale behind prepending prompts to the domain is to seamlessly integrate domain information without deteriorating the naturalness of the language model. Our findings suggest that augmenting prompts is more effective than using domain alone.

Prompt	AlignScore	FactCC	BARTscore↑	BS-FACT	ROUGE-L	BERTscore
w/o	60.48	22.02	-1.8033	88.88	35.81	91.50
keywords	60.45	21.94	-1.8039	88.88	35.76	91.49
topics	60.40	21.63	-1.8063	88.88	35.78	91.50
components	60.67	21.72	-1.8036	88.88	35.76	91.50
concepts	60.48	21.76	-1.8047	88.88	35.81	91.51
features	60.53	21.66	-1.8041	88.88	35.81	91.50
points	60.37	21.57	-1.8088	88.87	35.79	91.50
in summary	60.55	21.67	-1.8052	88.88	35.70	91.49
to be brief	60.33	21.58	-1.8032	88.88	35.81	91.50
last of all	60.42	21.53	-1.8035	88.88	35.80	91.50
when all is said and done	60.66	21.59	-1.8012	88.88	35.75	91.50
bringing up the rear	60.64	21.68	-1.8020	88.89	35.80	91.50
in short	60.67	21.63	-1.8035	88.88	35.78	91.51
in other words	60.71	21.71	-1.7988	88.88	35.80	91.51
that is to say	60.78	21.82	-1.7988	88.89	35.81	91.50
to rephrase it	60.66	21.96	-1.8011	88.89	35.80	91.50
take for example	60.76	21.87	-1.8025	88.88	35.81	91.50
to put it another way	60.45	21.69	-1.8013	88.89	35.76	91.49
case in point	60.62	21.81	-1.8033	88.87	35.81	91.51

Table 10: Results for each prompt, where the `domain` consists of three keywords. Adding "`that is to say`" to the three keywords yielded the best overall performance.

Method	**Text**
Domain	bia, falkirk, bi
Source	However, the Bairns boss has underlined that any forward signing will need to exhibit even more quality than two of his promising youngsters. "If I bring another striker in he's got to be better than young Botti **Bia**-**Bi** and Scott Shepherd," said Houston. "I would be looking for the more experienced type, and another defender would come in handy as well." Eighteen-year-old Bia-Bi, a London-born Scot who has progressed through **Falkirk**'s academy, glanced in a fine equalising header against Cowdenbeath on Saturday to ensure Houston's side left Central Park with a point...
PMI_{DC}	Falkirk manager Peter Houston has not ruled out bringing in a new striker in the January transfer window.
Gold	Peter Houston is still seeking to fine-tune his Falkirk squad, with a striker and defender pinpointed as priorities.

Table 11: **Case 1 error.** Inconsistent words are highlighted in *red* fonts. Extracted keywords may not fully reflect domains of source text. In this example, the domain should be more related to terms like *transfer* or *football* rather than specific names of individuals or institutions. Hence, terms closely related with transfer (such as *January*) were not adequately penalized.

Method	Text
Domain	invest, richest, investment
Source	The **investment** follows "several months of negotiations", a company statement to the Saudi stock exchange said. The prince, who is one of the world's **richest** men, owns stakes in many well-known companies, including News Corporation. He also has investments in a number of media groups in the Arab world. "Our investment in Twitter reaffirms our ability in identifying suitable opportunities to **invest** in promising, high-growth businesses with a global impact." Prince Alwaleed said.
PMI_{DC}	Saudi Arabia's Prince Alwaleed bin Talal has bought a 10% stake in Twitter in a deal worth $2bn (31.8bn).
Beam	Saudi Arabia's Prince Alwaleed bin Talal has agreed to buy a 10% stake in Twitter for $3bn (32.3bn).

Table 12: **Case 2 error.** Inconsistent words are highlighted in *red* fonts. The appropriate domain, but not properly regulated in accounting numbers. Hallucinations related to proper nouns, numbers and statistics, have long been significant issues in language models. Our approach could not completely address this issue.

Method	Text
Domain	claire, marathon, equestrian
Source	When **Claire** was told she would spend the rest of her life in a wheelchair after a spinal injury, she wanted to get back on her feet as quickly as possible and regain her independence. For the past three months she has been training intensively for the **marathon** using a robotic walking suit to prove she is just as determined as in her sporting days. ... former champion British **equestrian** Lucinda Green. "There's a lot of people who are worse off than me and haven't got the support I've got, so I want to raise as much as I can." But, when the marathon is over, Claire thinks that for the first time in six years, she will be delighted to return to her wheelchair.
PMI_{DC}	A paralysed equestrian rider is taking part in the London Marathon in a bid to become the first person in the world to walk unaided.
Beam	Claire Gwynne, who was paralysed from the chest down in 2006, is taking part in the London Marathon.

Table 13: **Case 3 error.** Inconsistent words are highlighted in *red* fonts. Constraints of domain-conditional term can prevent direct expressions, potentially leading to ambiguity and generating incorrect results. In this example, penalizing the domain term *Claire* led to the removal of the hallucinated term *Gwynne*. However, beyond this correction, the conveyed information remained somewhat inaccurate.