

LoKI: Money Laundering Report Generation via Logical Table-to-Text using Meta Learning

Ram Ganesh V*, Harika CM*, Rajesh Kumar Ranjan, Debasmita Das, Siddhartha Asthana

AI Garage, Mastercard, Gurgaon, India
{firstname.secondname}@mastercard.com

Abstract

The submission of Anti-Money Laundering (AML) reports is a key compliance mandate that financial institutions strictly adhere to, which involves a detailed scrutiny of accounts and transactions that may potentially be suspicious. Previous research have explored the automation of report creation via table-to-text generation. Our paper introduces a novel approach utilising table-to-text generation, known as Logical Key Inference (LoKI). Proposed method employs meta-learning to augment the content selection and addresses three salient gaps in logical table-to-text generation: (1) dynamic selection of pertinent table attributes, (2) mitigating hallucination during the attribute-to-value mapping process, and (3) the ability to incorporate newly added attributes without the demand for explicit additional training. Through experiments and ablation studies, we show that LoKI outperforms existing generative methods for compliance reporting.

1 Introduction

Money laundering is a global issue affecting nations in terms of financial health and physical security including terrorism, human trafficking etc. (IMF, 2023). According to an estimate (Kolmar, 2022), 2-5% of global GDP (\$2 trillion) is laundered in a single year. Hence, combating money laundering is more urgent than ever, especially for financial institutions like banks and payment networks. Financial institutions are required to provide highly confidential reports that include information on suspicious activities to national law enforcement authorities investigating financial matters. Financial analysts thoroughly review historical transactions regarding potentially risky accounts before scrupulously writing the reports. These reports are often written after analysing data stored in a tabular format containing suspicious accounts and corresponding transaction history (see

*These two authors contributed equally to this work

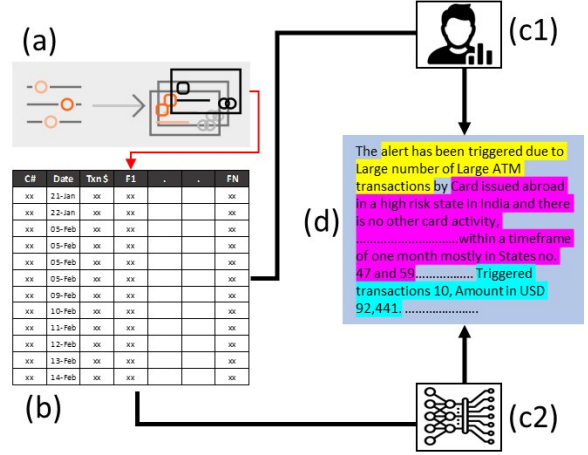


Figure 1: **AML report generation process** (a) Exploratory system to flag suspicious Money Laundering accounts and transactions (b) Tabular data containing historical transactions of the potential suspicious accounts (c) Report generation: (c1) Financial analyst generating AML reports by reviewing the tabular data (c2) Alternatively system-generated AML reports (d) The text report containing factual evidence against each suspicious account.

Figure 1). Research tried to reduce the manual effort by automating the process of report creation using Table-to-text generation methods (Lin et al., 2022).

Table-to-text generation (TTG) is an active area of research aiming to generate text from tabular data. Existing methods for TTG includes static attribute selection (i.e., rule-based (Reiter and Dale, 2000), and template based (Oh and Rudnicky, 2000)), conditional copy of text, (Puduppully et al., 2019), and fine-tuning sequence-to-sequence models (Kale and Rastogi, 2020). Even though the generated text is linguistically fluent, the challenge is that the surface-level spurious correlations are easier to capture than the causal relationship between tabular data and generated text, resulting in hallucination, generation of incorrect facts, and inconsistent content planning and document structuring. There are recent research work that ad-

dress content planning by using copy-mechanism (Gehrmann et al., 2018) or neural models based on soft-template (Wiseman et al., 2018). Researchers have also explored training a separate planning module to produce relevant content, which is then fed into a generator (Goldfarb-Tarrant et al., 2020) but the strategies result in a disconnection between planning and generation of high fidelity logically structured summaries. Given how sensitive and confidential nature of this data is using we cannot use the latest models released by OPEN AI like GPT 3.5 and GPT 4 due to privacy concerns.

To address the above gaps, we formulate three objectives in our proposed solution: (1) Dynamic selection of logical key table attributes (i.e. feature selection) to include in the report, (2) Ability to handle hallucination while mapping attributes to values (3) Adaptability of the system when new attributes get added with changing landscape of money laundering, without explicit additional training. Our proposed solution, LoKI, generates high-fidelity, logical, well structured, and insightful reports given the tabular data with significantly high number of attributes. The focal point of LoKI is Logical Key Inference - it adequately selects only the important, logically correlated attributes to generate the final report. LoKI uses Meta-Learning for content selection and explores BART and T5 for TTG. We perform the experiments on a repository of real-world financial transaction data for potential high-risk accounts related to money laundering. We show that the proposed method outperforms existing generative methods through extensive experiments and ablation studies.

2 Related Work

Natural text generation has been a significant focus of research across various fields such as sports (Wiseman et al., 2017), weather (Liang et al., 2009), and health (Lee, 2018) sectors. While advancements (Chen et al., 2020b) have been made, these methods often suffer from imprecision, hallucinations, and lack of proper content planning and structuring. To address these, researchers have proposed architectures (Castro Ferreira et al., 2019) that include multi-step processes (Tian et al., 2019; Puduppully and Lapata, 2021; See et al., 2017; Zeng et al., 2018; Liu and Lapata, 2018) such as discourse ordering, text structuring, and surface realization. However, most of these methods end up restating facts and producing relatively short texts.

The introduction of datasets like Logic2Text and LogicNLG shifted the focus towards ensuring the fidelity of logical-level generations. LOGEN (Zhang et al., 2021) and PLOG (Liu et al., 2022a) are two models that operate on these datasets, utilizing logical form and table content to generate target text. Although they improve generation fidelity, they require logic as an input, highlighting the need for selecting the most important logical relationships. A model introduced by (Zhao et al., 2023) in 2023 provided a solution for diversity and faithfulness in Logic2Text by using logical forms as mediators for controllable text generation. Despite the improvement, this model struggled to generate comprehensive paragraph summaries that capture various key perspectives in a table.

In response to these issues, (a) we propose a method to generate high-fidelity logical and insightful summaries given the table without any logic or cells as input (b) our model dynamically selects the significant features to appear in the reports and ignores the irrelevant pieces of information.

3 Proposed Solution

LoKI has three parts as shown in Figure 2. (a) ‘Logical Key’ attribute selection using Model Meta Learning, (b) TTG Transformers for generating well structured report templates containing selected attributes (c) Replacing the attributes in template with corresponding values to generate final report to minimize hallucination.

3.1 ‘Logical Key’ attribute selection using Meta Learning

First, we dynamically select ‘logical key’ table attributes for the report using a meta-learning algorithm, where each task governs the selection of a specific attribute. This approach overcomes the limitations of traditional machine learning methods, such as the need for large datasets, long model training time, compromised performance with increased target classes, and high operating costs. Meta-learning optimizes the learning process, enabling faster learning for new tasks with fewer examples. The report generation task involves a table T and a natural language target text Y . Table T , with M rows and C columns, records card profiles with each column representing an aggregated feature at the account level. The target text outlines the potential risk associated with the card.

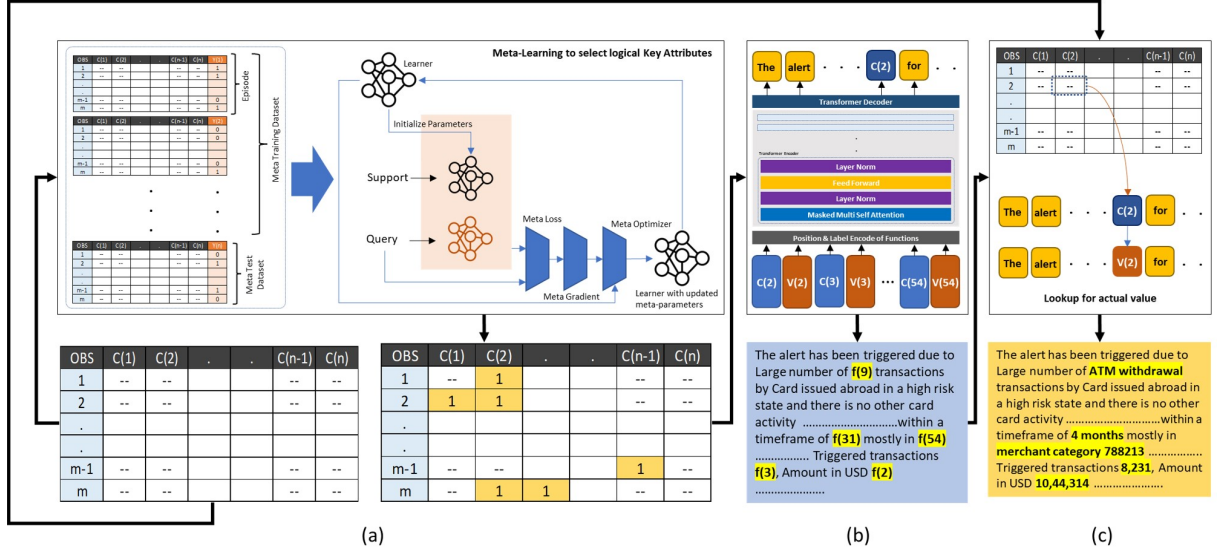


Figure 2: **LoKI** has three parts (a) Logical Key attribute selection using Meta Learning, (b) TTG Transformers for generating well structured report templates containing selected attributes (c) Replacing the attributes in template with corresponding values to generate final report to minimize hallucination.

3.1.1 Defining a Meta Learning Task

Every attribute present in the table is a meta learning task and the number of tasks correspond to the number of attributes available in the table. We used a supervised approach here. The reports are annotated (refer to methodology section) to create a target variable for every episode. For a specific episode, if the corresponding attribute is included in the report, then the target variable is 1, else it is set to 0. The same process is repeated for all the attributes to create the final set of tasks for meta-learning algorithm

3.1.2 Training a Meta Learning Algorithm

Let us consider a model f parameterized by a parameter θ , and let $p(N)$ represent a distribution over tasks. First, we randomly initialize the model parameter θ and sample a batch of tasks from the task distribution, denoted as $N_i \sim p(N)$. We then sample P different tasks, and the set of tasks can be represented as $N = N_1, N_2, N_3, \dots, N_P$.

For each task N_i in the set of tasks N , we sample k data points and prepare our training and test sets as follows:

$$D_i^{train} = (x_1, y_1), (x_2, y_2), \dots, (x_k, y_k) \quad (1)$$

$$D_i^{test} = (x_1, y_1), (x_2, y_2), \dots, (x_k, y_k) \quad (2)$$

Next, we employ a neural network model to train on the training set D_i^{train} and minimize the loss using gradient descent and obtain the optimal parameters θ'_i .

$$\theta'_i = \theta - \alpha \nabla_{\theta} L_{N_i}(f_{\theta}) \quad (3)$$

where $L_{N_i}(f_{\theta})$ denotes the cross-entropy loss function.

So for each of the tasks, we sample k data points and minimize the loss on the train set and get the optimal parameters. As we sampled P tasks we will have P optimal parameters $\{\theta'_1, \theta'_2, \theta'_3, \dots, \theta'_P\}$. Now, we perform meta optimization in the test set i.e D_i^{test} here we try to minimize the loss in the test set. We minimize the loss by calculating the gradient with respect to our optimal parameter θ'_i calculated in the previous step and update our randomly θ_i initialized parameter using our test set. It can be mathematically represented as :

$$\theta = \theta - \beta \nabla_{\theta} \sum_{N_i \sim p(N)} L_{N_i}(f_{\theta'_i}) \quad (4)$$

3.1.3 Getting the ‘Logical Key’ Attributes

The meta-learning algorithm assigns a probability score to each attribute for every account. If the score exceeds a predetermined threshold, we will carry forward that attribute and its respective value to the subsequent stage. Any attributes that do not meet this threshold are eliminated before

being processed by the language model. It’s worth noting that the number of selected attributes can vary between accounts.

3.2 Table to Text Generation using Transformers

To generate well structured report templates containing selected attributes we have used transformer-based table to text generators. Theoretically any language model can achieve the knowledge transfer from the table-to-text downstream task. To evaluate this, we have included several language model such as GPT-2 ((Radford et al., 2019)), BART-Base ((Lewis et al., 2019b)), BART-Large CNN ((Lewis et al., 2019a)), T5 Base ((Raffel et al., 2020)). The output of this step is saved as template report containing dynamically selected attributes.

3.3 Mitigating Factual Hallucination

In this step, we assign the attributes in the template report, created in the prior stage, to the actual values derived from the aggregated transaction data. This involves conducting a lookup for the original values of the selected attributes. We substitute the function placeholders with these actual values to produce the final report.

4 Methodology

In this section, we provide a synopsis of the different experiments that we have performed followed by analysis of the results for in-depth insights and deliberations.

4.1 Data Description

We conducted experiments on a three-year real-world financial dataset of potentially high-risk accounts tied to money laundering, covering over 20,000 accounts. To ensure privacy, we concealed customers’ identities and specific experiment timeframes. We aggregated attributes at the account level, examining merchant, transaction, card, and other features such as origin country, transaction volume, transaction channels, card type, and suspicious activities.

4.2 Annotation

We perform the experiments on historical reports prepared by the analyst. We manually annotate the reports to evaluate the inclusion of specific attributes. The numbers/facts are replaced with their corresponding attributes.

4.3 Experiment Protocols

Our study focuses on generating highly accurate, logically correct and precise, well-structured report generation from tabular data. We evaluated the performance of LoKI (1) quantitatively by comparing it with existing generative methods on metrics like BLEU-4, ROUGE-4 , Precision, Recall, etc and (2) qualitatively with the help of internal experts.

4.4 Implementation Details

This section outlines the baseline models including LLaMa (Touvron et al., 2023), GPT-2 (Radford et al., 2019), BART-Base (Lewis et al., 2019b), BART-Large CNN (Lewis et al., 2019a), T5-Base (Raffel et al., 2020) and our custom models: Meta-learning+BART-Large and Meta-learning+T5-Base in a fully-supervised setting. Using Transformers and PyTorch, we conducted numerous experiments, setting the max length to 6000 for source and 2000 for target sequences. Each model was fine-tuned for ten epochs with the entire training data. We omitted logical table-to-text methods like PLOG and LOFT from our experiments as they require a logical input form. In contrast, our pipeline doesn’t require such input and autonomously selects logical statements based on attribute values. We evaluated our models (Liu et al., 2022a; Lin et al., 2022) using surface-level metrics (BLEU-4, ROGUE-4, Precision, Recall) and logic-based metrics (SP-Acc, TAPEX-Large (Liu et al., 2022b)) to ensure logical fidelity in output. Evaluation was based on n-gram matching between model-generated summaries and references. All these were implemented with the Tensorflow and PyTorch.

5 Result

In this section we present, quantitative and qualitative (human) evaluation of ‘LoKI’.

(1) Quantitative Evaluation: The experimental results are detailed in Table 1. Despite GPT-2’s extensive training, it struggles with numeric data, dates, and logical patterns. Our Meta-learning models outperform both T5-Base and BART-Large, indicating that focusing on selective features enhances output quality and fidelity. The consistent advancement across various models underscores the universal applicability of our approach. (see Table 1).

Surface-Level Evaluation					Logical Fidelity		
Models	BLEU-4	ROUGE-4	Precision	Recall	F1	SP-Acc	Tapex-Acc
GPT 2	27.1	23.5	30.1	25.4	27.6	9.4	12.5
Bart-Base	40.5	43.9	54.5	46.7	50.3	42.1	35.7
BART-Large CNN	53.7	54.2	59.5	72.5	65.4	51.9	64.3
T5-Base	58.6	59.2	60.8	63.7	61.8	67.5	59.9
LLaMa	43.1	46.5	58.1	60.5	59.2	65.4	68.1
(ML*+BART)							
(-Large CNN)	76.4	75.6	79.9	87.5	83.5	80.9	83.2
(ML*+T5-Base)	79.7	78.3	82.7	84.4	83.6	84.7	89.8

Table 1: Evaluation results for all baselines and our (marked with *) models. We use meta learning and T5 Large as a language model to produce the results

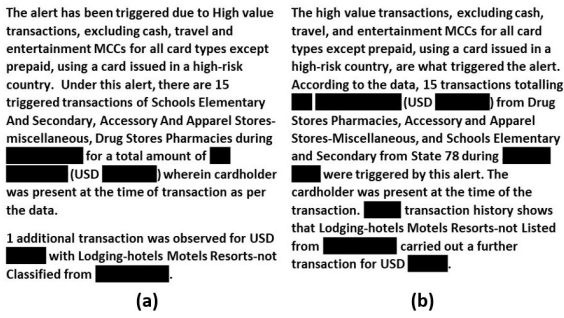


Figure 3: **Human Evaluation** (a) The report manually generated by the analyst (b) The report generated by LoKI. (Pursuant to internal and legal controls to protect data, confidentiality, and privacy, some of the parts of reports are blackened out, so that the results cannot be traced back to any original transaction or customers.)

(2) Qualitative Evaluation We randomly sampled 400 examples from the test data. Internal experts scored each generated summary in the discrete range between 0 and 4 with the help of according to criteria adopted in (Chen et al., 2020a). Non-sense (0): the sentence suffers with respect to text fluency and logic fidelity, and people need help understanding its meaning. Wrong (1): the sentence is fluent, but it contains factual errors, i.e. accuracy of recalled risk is low. Partially correct (2): the sentence describes multiple facts. Most of them are wrong, but it contains at least one factually correct sentence. Almost Correct (3): the sentence describes multiple facts. Most of them are correct, but it contains at least one factual error. Entirely Correct (4): the sentence is of high quality in fluency and risk accuracy. The evaluation is based on the generated summaries and the tabular input data. From the evaluation results, the proposed models *ML+T5-Base* and *ML+BART-Large* are the highest scorers with 89% and 84% of entirely correct statements, respectively.

6 Conclusion

In this work, we formulate the problem of logical key feature selection from a tabular dataset consisting of significantly large number of attributes to generate a structured, factually accurate and logically correct Anti-Money Laundering report for suspicious accounts. In our case, we emphasized that the facts presented in the report must be unequivocally accurate. Inclusion of incorrect facts in the generated report can potentially lead to legal and reputational harm. To achieve this, we are using the natural language model to learn only the template of the report, unlike previous research where the language model generates the final report. Through our results, we demonstrate that proposed method considerably outperforms the existing state-of-art models. We believe that our work can highly successful in convincing regulators about the applications of AI in related domains as well.

References

- Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. [Neural data-to-text generation: A comparison between pipeline and end-to-end architectures](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 552–562, Hong Kong, China. Association for Computational Linguistics.
- Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, and William Yang Wang. 2020a. [Logical natural language generation from open-domain tables](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7929–7942, Online. Association for Computational Linguistics.

- Zhiyu Chen, Harini Eavani, Wenhui Chen, Yinyin Liu, and William Yang Wang. 2020b. [Few-shot NLG with pre-trained language model](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 183–190, Online. Association for Computational Linguistics.
- Sebastian Gehrmann, Falcon Dai, Henry Elder, and Alexander Rush. 2018. [End-to-end content and plan selection for data-to-text generation](#). In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 46–56, Tilburg University, The Netherlands. Association for Computational Linguistics.
- Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020. [Content planning for neural story generation with aristotelian rescoring](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4319–4338, Online. Association for Computational Linguistics.
- IMF. 2023. International monetary fund. <https://www.imf.org/en/About/Factsheets/Sheets/2023/Fight-against-money-laundering-and-terrorism-financing>. [Online; accessed 17-feb-2023].
- Mihir Kale and Abhinav Rastogi. 2020. [Text-to-text pre-training for data-to-text tasks](#). In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 97–102, Dublin, Ireland. Association for Computational Linguistics.
- Chris Kolmar. 2022. 20 MONEY LAUNDERING STATISTICS [2023] FACTS ABOUT MONEY LAUNDERING IN THE U.S. <https://www.zippia.com/advice/money-laundering-statistics/>. [Online; accessed 17-feb-2023].
- Scott H Lee. 2018. Natural language generation for electronic health records. *NPJ digital medicine*, 1(1):63.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019a. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019b. [BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension](#). *CoRR*, abs/1910.13461.
- Percy Liang, Michael I Jordan, and Dan Klein. 2009. Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 91–99.
- Xiexiong Lin, Huaisong Li, Tao Huang, Feng Wang, Linlin Chao, Fuzhen Zhuang, Taifeng Wang, and Tianyi Zhang. 2022. A logic aware neural generation method for explainable data-to-text. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3318–3326.
- Ao Liu, Haoyu Dong, Naoaki Okazaki, Shi Han, and Dongmei Zhang. 2022a. [PLOG: Table-to-logic pre-training for logical table-to-text generation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5531–5546, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022b. [TAPEX: Table pre-training via learning a neural SQL executor](#). In *International Conference on Learning Representations*.
- Yang Liu and Mirella Lapata. 2018. Learning structured text representations. *Transactions of the Association for Computational Linguistics*, 6:63–75.
- Alice Oh and Alexander Rudnicky. 2000. Stochastic language generation for spoken dialogue systems. In *ANLP-NAACL 2000 Workshop: Conversational Systems*.
- Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6908–6915.
- Ratish Puduppully and Mirella Lapata. 2021. Data-to-text generation with macro planning. *Transactions of the Association for Computational Linguistics*, 9:510–527.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- 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):1–67.
- Ehud Reiter and Robert Dale. 2000. Building natural generation systems. *Studies in Natural Language Processing*. Cambridge University Press.
- Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.
- Ran Tian, Shashi Narayan, Thibault Sellam, and Ankur P Parikh. 2019. Sticking to the facts: Confident decoding for faithful data-to-text generation. *arXiv preprint arXiv:1910.08684*.

- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Sam Wiseman, Stuart M Shieber, and Alexander M Rush. 2017. Challenges in data-to-document generation. *arXiv preprint arXiv:1707.08052*.
- Sam Wiseman, Stuart M Shieber, and Alexander M Rush. 2018. Learning neural templates for text generation. *arXiv preprint arXiv:1808.10122*.
- Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 506–514.
- Ningyu Zhang, Hongbin Ye, Jiacheng Yang, Shumin Deng, Chuanqi Tan, Mosha Chen, Songfang Huang, Fei Huang, and Huajun Chen. 2021. Logen: few-shot logical knowledge-conditioned text generation with self-training. *arXiv preprint arXiv:2112.01404*.
- Yilun Zhao, Zhenting Qi, Linyong Nan, Lorenzo Jaime Yu Flores, and Dragomir Radev. 2023. Loft: Enhancing faithfulness and diversity for table-to-text generation via logic form control. *arXiv preprint arXiv:2302.02962*.