AkibaNLP-TUT: Injecting Language-Specific Word-Level Noise for Low-Resource Language Translation

Shoki Hamada[1] **Tomoyoshi Akiba**[1] **Hajime Tsukada**[2]
[1]Toyohashi University of Technology [2] Aichi Sangyo University
{hamada.shoki.ew, akiba.tomoyoshi.tk}@tut.jp
tsukada@asu.ac.jp

Abstract

In this paper, we describes our system for the WMT 2025 Low-Resource Indic Language Translation Shared Task. The language directions addressed are Assamese↔English and Manipuri→English. We propose a method to improve translation performance from low-resource languages (LRLs) to English by injecting Language-specific word-level noise into the parallel corpus of a closely related high-resource language (HRL). In the proposed method, word replacements are performed based on edit distance, using vocabulary and frequency information extracted from an LRL monolingual corpus. Experiments conducted on Assamese and Manipuri show that, in the absence of LRL parallel data, the proposed method outperforms both the w/o noise setting and existing approaches. Furthermore, we confirmed that increasing the size of the monolingual corpus used for noise injection leads to improved translation performance.

1 Introduction

There are approximately 7,000 languages in the world, but only a small subset of high-resource languages (HRLs) have sufficiently developed parallel corpora for machine translation (MT). For these languages, research leveraging few-shot learning with parallel data (Zhu et al., 2023) and large-scale multilingual language models (mLLMs) (Xu et al., 2023; Zhou et al., 2023) has progressed. Such efforts have enabled the learning of shared cross-lingual embedding spaces, thereby facilitating cross-lingual transfer.

In contrast, many languages are low-resource languages (LRLs) for which only monolingual data is available. Compared to parallel data, monolingual data is easier to collect and is widely used for techniques such as back-translation (BT)(Sennrich et al., 2016a) and continued pretraining of mLLMs. This study aims to improve LRL→English translation accuracy by leveraging LRL monolingual data to inject language-specific, word-level noise into parallel data of closely related HRLs.

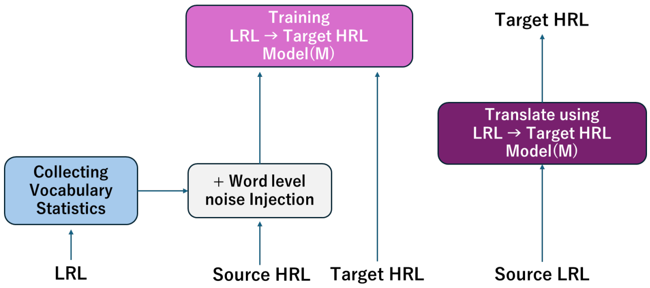

Figure 1: Overview of our proposed method: Language-Specific Word-Level Noise Injection

Maurya et al. (2024) proposed CharSpan, a method that injects character-level noise into HRL parallel data with high lexical similarity to a LRL. This method improves translation accuracy for the LRL by leveraging its character list. However, this approach does not sufficiently capture word-level statistical characteristics such as the LRL's vocabulary frequency and distribution.

To address this limitation, we propose a method that uses word lists and frequency information extracted from LRL monolingual data to add word-level noise to HRL parallel data. We further analyze the effect of this method under conditions where LRL parallel data is available versus unavailable. Experimental results show that when using only LRL monolingual data, our method outperforms existing approaches. In contrast, when LRL parallel data is available, the performance gap with existing methods is small. We also observe that increasing the size of the monolingual data used for noise injection tends to improve performance, and that including the test dataset in the monolingual data yields additional performance gains.

2 Related Work

Some studies have examined the effects of adding noise to parallel data on its diversity and the ro-

Proceedings of the Tenth Conference on Machine Translation, pages 1259–1264
November 8-9, 2025 ©2025 Association for Computational Linguistics

bustness of translation models, but the impact on cross-lingual transfer has not been thoroughly investigated. Gal and Ghahramani (2016) proposed Word Dropout, which randomly sets some word embeddings to zero vectors. Wang et al. (2018) proposed a data augmentation method that adds random word replacements to parallel data. While both of these methods enhance the diversity of parallel data, their effectiveness in improving cross-lingual transfer capability is limited.

A representative data augmentation technique that leverages monolingual data in neural machine translation is back-translation (BT) (Sennrich et al., 2016a). When parallel data is scarce, BT generates pseudo-parallel data by using target-side monolingual data and a reverse-direction translation model. More recently, iterative back-translation (IBT) (Morita et al., 2018; Hoang et al., 2018; Zhang et al., 2018) has been proposed, which extends BT in both directions. IBT utilizes monolingual data from both sides to generate pseudo-parallel data in both directions and iteratively alternates between generating this data and updating the translation models in both directions.

3 Method

In this chapter, we propose a method to enhance robustness in LRL→En translation and promotes cross-lingual transfer by adding LRL-specific word-level noise into a parallel corpus of a closely related HRL. The noise consists solely of word replacements, where the edit distance is selected based on a geometric distribution. The replacement candidates are chosen using frequency-weighted selection, thereby injecting LRL words into the related HRL.

Specifically, we use approximately 160,000 sentences of Assamese monolingual text to add noise to Bengali–English parallel data. Figure 1 shows an overview of the proposed method. An example of the noise injection process is illustrated in Figure 2. Furthermore, by using the model trained on the noise-injected parallel data as a back-translation model, we perform En→LRL translation.

3.1 Language-Specific Word-level noise

We add word-level noise to the source-side training data D_{HRL} of the HRL pair to create the noisy parallel data D'_{HRL}.

First, we randomly select a word index x_i from any given sentence x. Next, we determine the edit

HRL(Bn):	এমন কথা কিন্তু তিনি আপনাকে কোনওদিনও বলেননি ।
Eng:	But he never told me.
HRL(Bn) + Noise:	এমন কথা সিন্ধু তিনি আপনাকে কোনোদিনে বলেননি ।

Figure 2: Example of word-level noise injection for Bengali (HRL). The original Bengali sentence and its English sentence are shown at the top. In the noisy version (bottom), Bengali words are replaced with words selected from an Assamese vocabulary list.

distance d according to Equation 1, where p is the success probability of the geometric distribution:

$$P(d = k) \propto p(1 - p)^{k-1} \quad (k = 1, 2, \ldots, K) \qquad (1)$$

The parameters K and p control the distribution of noise magnitude. Then, based on the edit distance between a candidate word w and x_i, we extract a candidate set $V(d, x_i)$ from the LRL vocabulary V_{LRL}:

$$V(d, x_i) = \{w | w \in V_{LRL}, ED(w, x_i) = d\} \qquad (2)$$

Here, $ED(\cdot, \cdot)$ denotes the Levenshtein distance. The replacement word w' is selected according to the product of $P(d)$ and the relative word frequency $f(w')$ in the LRL monolingual corpus:

$$P(w') = P(d) \cdot \frac{f(w')}{\sum_{w \in V(d, x_i)} f(w)} \qquad (3)$$

This procedure is repeated until the proportion of characters changed by substitution in each sentence reaches a predefined target ratio.

3.2 Back translation

In this study, we apply translation model M, trained on HRL parallel data D'_{HRL} augmented with word-level noise, to LRL-to-English translation. Specifically, to perform EN→LRL translation, we first translate the LRL monolingual data into English using M, thereby creating pseudo-parallel data $\hat{D}_{\text{LRL-EN}}$.

Subsequently, this pseudo-parallel data is used as input to train an English→LRL translation model.

4 Experimental Setup

4.1 Datasets

We target Bengali (Bn) as the HRL and Assamese (As) and Manipuri (Mni) as the LRLs, using multiple parallel and monolingual corpora for model

Corpora	Language	Usage	# Sentences
Samanantar	English-Bengali	Train	8,604,580
WMT25 Shared Task	English-Assamese	Train / Valid	Train: 53,003 Valid: 997
	English-Manipuri	Train / Valid	Train: 22,690 Valid: 997
FLORES-200	English-(Bengali, Assamese, Manipuri)	Valid / Test	Valid: 977 Test: 1,012
Community 2017 Wikipedia 2021	Assamese	Noise Injection	63,627 100,000

Table 1: Overview of the parallel and monolingual corpora used in this study, including the languages, their intended usage, and the number of sentences.

training and evaluation. Table 1 provides an overview of the corpora used. For parallel corpora, we used the large-scale English–Bengali Samanantar corpus(Ramesh et al., 2022) and the English–Assamese and English–Manipuri parallel datasets provided by the WMT25 shared task(Pal et al., 2023; Pakray et al., 2024). For evaluation, we used the validation and test sets of FLORES-200(Costa-Jussà et al., 2022). Additionally, Assamese Wikipedia 2021 and Community 2017 were used as monolingual corpora to extract vocabulary for noise injection in the proposed method.

4.2 Data Preprocessing

For English data, we first perform Unicode normalization (NFKC) and applied tokenization using `sacremoses`. Next, to reduce case variation at sentence beginnings and in proper nouns, we applied truecasing, and finally, we learned and applied Byte Pair Encoding (BPE)(Sennrich et al., 2016b; Gage, 1994) using `subword-nmt`. The number of BPE merge operations was set to 16,000.

For Bengali, Assamese, and Manipuri data, we applied the same preprocessing steps as for English—NFKC normalization, tokenization with `sacremoses`, and BPE (16,000 merges)—but did not apply truecasing.

4.3 Settings

For the word-level noise injection, the maximum edit distance K was set to 5, and the actual edit distance was sampled from a geometric distribution with a success probability $p = 0.5$. Noise was injected to each sentence until the proportion of characters altered by substitution reached the target ratio of 10%. Figure 4 shows the list of replacement characters used in CharSpan.

Parameter	Value
Architecture	Transformer (Encoder 6 layers / Decoder 6 layers)
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.98$)
Initial learning rate	$5e-4$
LR scheduler	Inverse Sqrt Decay
Gradient clip norm	1.0
Dropout	0.2
Max tokens / batch	8,000
Early-stopping patience	5 validations
GPU	2 × NVIDIA GeForce RTX 2080 Ti

Table 2: Model implementation and training details

We adopted the Transformer architecture (Vaswani et al., 2017) as the translation model. Both the encoder and decoder consisted of 6 layers, and optimization was performed using Adam (Kingma and Ba, 2014) with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.98$. The initial learning rate was set to 5×10^{-4}, with Inverse Sqrt Decay as the learning rate scheduler. To prevent gradient explosion, the gradient clip norm was set to 1.0. The dropout rate was set to 0.2, and the maximum number of tokens per batch was 8,000. Training employed early stopping, terminating when the validation loss did not improve for 5 consecutive evaluations. Experiments were conducted using two NVIDIA GeForce RTX 2080 Ti GPUs. Table 2 presents the key hyperparameter settings used in our experiments.

4.4 Evaluation Metrics

For validation and evaluation, we used the official FLORES-200 dev/test sets. BLEU (Papineni et al., 2002) and chrF++ (Popović, 2017) were adopted as evaluation metrics.

5 Result and Analisys

5.1 Main Results: LRL→EN

Table 3 presents the translation results from LRL to English. **Bold** indicates the highest score in each setting. In the setting without LRL parallel

Models	As→En		Mni→En	
	BLUE	chrF	BLUE	chrF
w/o noise	5.49	25.2	1.29	**18.9**
CharSpan	10.44	36.1	**0.75**	18.3
Word-Level noise	**12.92**	**38.1**	0.65	17.3
w/o noise + parallel	19.07	44.4	9.8	34.8
CharSpan + parallel	**21.46**	**47.4**	11.97	**38.8**
Word-Level noise + parallel	21.44	46.9	**12.12**	37.3

Table 3: Experimental results of LRL→English translation with and without LRL parallel data.

data, the proposed method outperformed both the w/o noise and CharSpan baselines for Assamese across all evaluation metrics. This improvement is likely due to the effective utilization of vocabulary and word frequency distributions derived from Assamese monolingual data. In contrast, for Manipuri, the proposed method underperformed compared to both baselines.

When LRL parallel data was used, the proposed method outperformed w/o noise for both languages, but achieved only comparable gains to CharSpan. This suggests that when w/o noise already possesses a moderate level of translation capability, the improvements brought by noise injection may be limited. Furthermore, for Manipuri, the presence or absence of LRL parallel data resulted in differing levels of improvement, indicating that the proposed method is effective when the w/o noise already has a certain degree of translation capability.

5.2 Effects of Monolingual Data Size and Domain for Noise Injection

The noise injection function used in this study (Equation 3) extracts replacement candidates from the LRL vocabulary with frequency weighting. Expanding the size of the monolingual corpus increases the likelihood of selecting more informative candidates. To verify this effect, we conducted experiments in which the amount of monolingual data was restricted. Starting from the full Assamese monolingual pool, we create down-sampled subsets with the following sentence counts (vocabulary sizes): *120k* (154,461), *100k* (139,751), *50k* (93,797) and *10k* (33,620). Additionally, we evaluate a setting where the full monolingual data is augmented with the Assamese test sentences from FLORES-200 (*full data + test*). For each subset, we learn BPE, train the model with the same configuration as described in Table 2, and evaluate it on FLORES-200.

The results are shown in the figure 3. The

	# Sentences	Vocabulary Size
(full data + test)	164,639	182,727
(full data)	163,627	180,810
	120,000	154,461
	100,000	139,751
	50,000	93,797
	10,000	33,620

Table 4: Monolingual Assamese subsets used to build the LRL vocabulary for noise injection. Vocabulary size counts unique types after preprocessing.

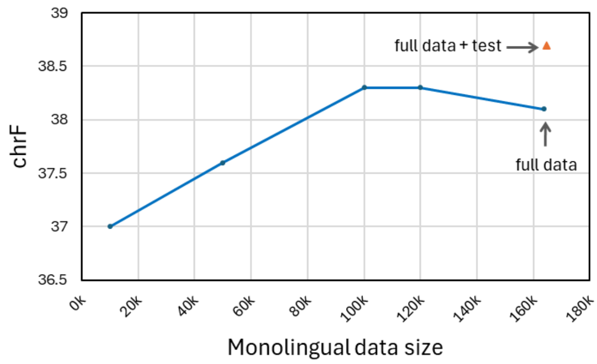

Figure 3: Effect of Assamese monolingual data size and domain on chrF scores for Assamese→English translation.

full data + test setting achieved the highest score of 38.7, slightly surpassing the 38.1 of the full data setting. The 120k and 100k settings both yielded similar performance at 38.3, while 50k achieved 37.6 and 10k scored 37.0, showing a gradual decline in performance as the amount of data decreased. These results suggest that increasing the size of the monolingual data makes it easier to select more informative replacement candidates during noise injection, potentially leading to improved translation performance. Furthermore, in the full data + test setting, including sentences from the same domain as the evaluation data in the monolingual corpus may have contributed to the performance improvement.

5.3 Shared Task Results

Table 5, presents the evaluation results of LRL↔English translation on the test set provided in the shared task. The evaluation metrics are BLEU, METEOR(Banerjee and Lavie, 2005), ROUGE-L(Lin, 2004), chrF, and TER(Snover et al., 2006). The translation directions are Assamese→English, English→Assamese, and Ma-

Direction	BLEU↑	METEOR↑	ROUGE-L↑	chrF↑	TER↓
Assamese→English	12.28	0.54	0.56	55.61	78.24
English→Assamese	14.03	0.38	0.01	53.76	74.08
Manipuri→English	5.74	0.33	0.37	41.28	109.95

Table 5: Evaluation results of LRL↔English translation using the test set provided in the shared task.

Language	Script	Candidate Alphabets
Assamese	Bengali	'◌ঁ', '◌ং', '◌ঃ', 'অ', 'আ', 'ই', 'ঈ', 'উ', 'ঊ', 'ঋ', 'এ', 'ঐ', 'ও', 'ঔ', 'ক', 'খ', 'গ', 'ঘ', 'ঙ', 'চ', 'ছ', 'জ', 'ঝ', 'ঞ', 'ট', 'ঠ', 'ড', 'ঢ', 'ণ', 'ত', 'থ', 'দ', 'ধ', 'ন', 'প', 'ফ', 'ব', 'ভ', 'ম', 'য', 'ল', 'শ', 'ষ', 'স', 'হ', '◌়', '◌া', '◌ি', '◌ী', '◌ু', '◌ূ', '◌ৃ', '◌ে', '◌ৈ', '◌ো', '◌ৌ', '◌্', 'ৎ', 'ড়', 'ঢ়', 'য়', '০', '১', '২', '৩', '৪', '৫', '৬', '৭', '৮', '৯', 'ৰ', 'ৱ'

Figure 4: Candidate alphabets for Assamese in the Bengali script, used as noise for CharSpan.

nipuri→English.

Our system was particularly designed for the Assamese→English direction. The performance gap between Assamese and Manipuri reflects the disparity in the availability of monolingual and parallel data, as well as differences in vocabulary and grammar with Bengali. These results indicate that the proposed method is especially effective for LRLs that are lexically close to the corresponding HRL.

6 Conclusion

In this study, we proposed Language-specific word-level noise injection method for the parallel corpus of a HRL closely related to a LRL, using vocabulary and frequency information extracted from the LRL's monolingual data. Experiments on Assamese and Manipuri demonstrated that, particularly in the absence of parallel data, the proposed method outperforms both the w/o noise setting and existing approaches. Furthermore, we showed that increasing the size of the monolingual data contributes to improved translation performance.

Limitations

First, the proposed method targets only a limited set of LRLs (Assamese and Manipuri), and its applicability to other languages has not been verified. In particular, it may be difficult to apply the method to LRLs whose script systems and lexical structures differ substantially from those of the HRL. Furthermore, the selection of replacement candidates for noise injection relies on word frequency and edit distance derived from the monolingual corpus of the LRL, and the potential for performance improvement may be limited depending on the scale of these statistics and the domain of the corpus.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Marta R Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Philip Gage. 1994. A new algorithm for data compression. *C Users J.*, 12(2):23–38.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. *Advances in neural information processing systems*, 29.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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.

Kaushal Maurya, Rahul Kejriwal, Maunendra Desarkar, and Anoop Kunchukuttan. 2024. CharSpan: Utilizing lexical similarity to enable zero-shot machine translation for extremely low-resource languages. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 294–310, St. Julian's, Malta. Association for Computational Linguistics.

Tomohiro Morita, Tomoyosi Akiba, and Hajime Tsukada. 2018. A study on unsupervised adaptation of neural machine translation with bidirectional back-translation (in Japanese). In *IPSJ SIG Technical Report*, volume 2018-NL-238, pages 1–5.

Partha Pakray, Santanu Pal, Advaitha Vetagiri, Reddi Krishna, Arnab Kumar Maji, Sandeep Dash, Lenin Laitonjam, Lyngdoh Sarah, and Riyanka Manna. 2024. Findings of WMT 2024 shared task on low-resource Indic languages translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 654–668, Miami, Florida, USA. Association for Computational Linguistics.

Santanu Pal, Partha Pakray, Sahinur Rahman Laskar, Lenin Laitonjam, Vanlalmuansangi Khenglawt, Sunita Warjri, Pankaj Kundan Dadure, and Sandeep Kumar Dash. 2023. Findings of the WMT 2023 shared task on low-resource Indic language translation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 682–694, Singapore. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2017. chrf++: words helping character n-grams. In *Proceedings of the second conference on machine translation*, pages 612–618.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 Indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. SwitchOut: an efficient data augmentation algorithm for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 856–861, Brussels, Belgium. Association for Computational Linguistics.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*.

Zhirui Zhang, Shujie Liu, Mu Li, Ming Zhou, and Enhong Chen. 2018. Joint training for neural machine translation models with monolingual data. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2023. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Multilingual machine translation with large language models: Empirical results and analysis. *arXiv preprint arXiv:2304.04675*.