Pushing the Limits of Low-Resource NER Using LLM Artificial Data Generation

Joan Santoso[1*] **Patrick Sutanto**[2*] **Billy Kelvianto Cahyadi**[2] **Esther Irawati Setiawan**[1]
Department of Informatics, Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)
[1]{joan, esther}@istts.ac.id
[2]{patrick.s21, billy_k20}@mhs.istts.ac.id

Abstract

Named Entity Recognition (NER) is an important task, but to achieve great performance, it is usually necessary to collect a large amount of labeled data, incurring high costs. In this paper, we propose using open-source Large Language Models (LLM) to generate NER data with only a few labeled examples, minimizing the need for extensive human-annotated data. Our proposed method is simple and can perform well using only a few labeled data points. Experimental results on diverse low-resource NER datasets show that our proposed data generation method can significantly improve the baseline. Additionally, our method can be used to augment datasets with class-imbalance problems and consistently improves model performance on macro-F1 metrics.

1 Introduction

Named Entity Recognition (NER) is a Natural Language Processing (NLP) task that classifies named entities in text, such as persons, organizations, locations, and more. NER is an important problem to solve and has a lot of use cases, such as relation extraction (Miwa and Bansal, 2016), question answering (Raiman and Miller, 2017), and a lot of other applications (Nallapati et al., 2016; Le and Titov, 2018; Banerjee et al., 2019).

Most work on NER needs access to a large number of labeled examples (Devlin et al., 2018; Wang and Lu, 2020; Wang et al., 2020) to perform well. Using distant supervision can alleviate the problem (Liang et al., 2020; Kim et al., 2022; Ren et al., 2015), but it usually needs domain expertise or a complex pipeline, and the results are also sensitive to the existence of a clean validation example (Zhu et al., 2023). Even recent work on few-shot NER usually assumes the existence of a dataset on the source task, which can be used to adapt to novel tasks (Fang et al., 2023; Zhou et al., 2023). Collecting a large amount of high-quality data for the NER task is very difficult and expensive, thus limiting the applicability of the NER model

Data augmentations have been successfully applied to address the data scarcity problem in NLP (Feng et al., 2021). However, their application to sequence labeling tasks is currently limited to simple heuristic approaches (Dai and Adel, 2020) or methods like back-translation (Yaseen and Langer, 2021), which still require a moderate amount of data and lack the diversity needed to generalize effectively on very limited amounts of data.

Recently, LLM have demonstrated impressive capabilities in performing various tasks (Brown et al., 2020; Ouyang et al., 2022; Schick et al., 2023) and generating data for tasks like classification (Chung et al., 2023), but their application on data augmentation in NER task is still limited. One approach use LLM to paraphrase text (Sharma et al., 2023), but this still lacks the necessary diversity. Another approach utilized entities from a predefined list to create NER data using LLM (Tang et al., 2023). We believe that this approach has several limitations, including the need for an iterative prompt engineering and a substantial volume of entity list, which may not be practical.

In this study, we tackle few-shot and low-resource NER by leveraging LLM without using a source task or unlabeled data. For cost-effectiveness and applicability, we utilize the GPTQ-quantized SOTA open-source model Mixtral (Jiang et al., 2024; Frantar et al., 2022). We show that our simple method boosts performance on various low-resource datasets. Notably, we observe a substantial 19-point increase in F1 score when using 1% data of the WNUT and Tweebank datasets. Additionally, our method improves macro-F1 scores on resource-scarce, class-imbalanced datasets, with a 4-point average increase on CrossNER.

[*]Equal contribution.

Findings of the Association for Computational Linguistics ACL 2024, pages 9652–9667

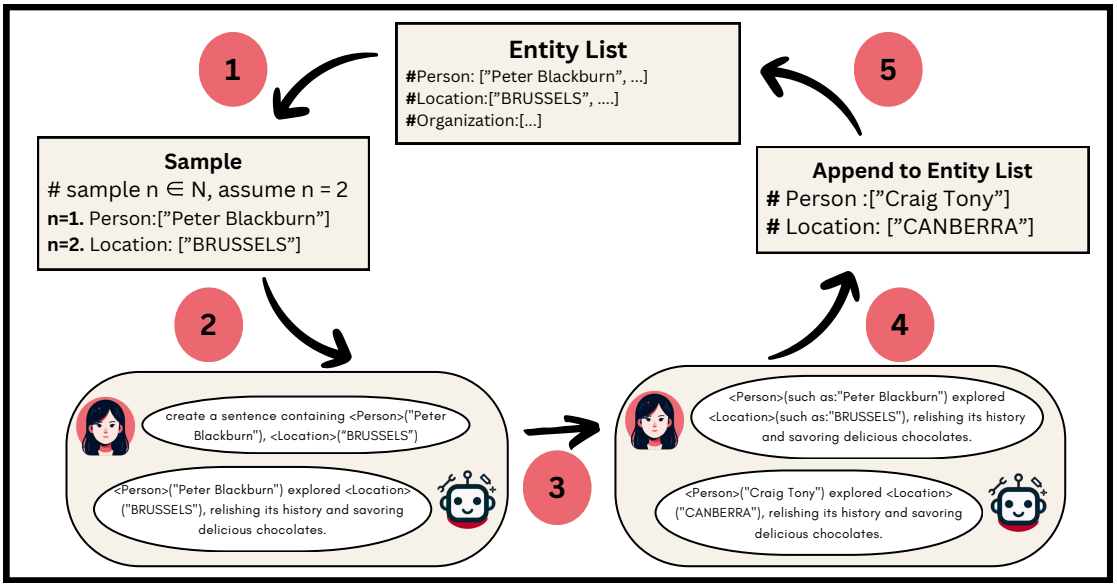

Figure 1: Proposed generation process

2 NER Data Generation using LLM

In this section, we introduce our method to enhance NER performance in low-data scenarios. The core concept involves generating sentences from entities randomly selected from a list, masking the sentence, subsequently using these to update the entity list. During data generation, we store all entities available in the dataset as E, with E_t representing all entities of a specific type. The process begins by initially generating sentences given certain entities, as explained in section 2.1, and then diversifying the entity list by masking entities and regenerating sentences, as detailed in section 2.2.

2.1 Generating Sentence

To generate a sentence, we first sample n entities uniformly from 0 to N, where N is the maximum number of entities in a sentence. For each entity, we randomly sample an entity type t (e.g., Person, Location). We then sample one entity from E_t, producing n entities with diverse types.

After sampling n entities, we prompt the LLM to generate a sentence given the sampled entity and its type. The prompt used for sentence generation involves 5 examples randomly sampled from the dataset as few-shot prompts. We format the entity as <Type>("Entity") to provide the LLM with type information. A simplified example is illustrated in Figure 1. We show the full prompt on the Figure 3 on Appendix D

To address the issue of LLM generating named entities that are not tagged, we relabel the tokens based on the entity list E. We modify the label of tokens that have no label but exist in E. Additionally, we remove sentences that do not align with our template and cannot be parsed. During the sentence generation process, we also exclude sentences containing newline characters to ensure each training data only contains one sentence.

2.2 Diversifying Entity

As we assume a limited amount of data, the potential entity sample space is constrained. To address this, we utilize the LLM to generate similar entities based on sentences generated previously. Randomly generating entities may introduce entities that are out of domain. To overcome this, we provide information about the type and the previously generated entity to the LLM, asking it to generate a different entity. We mask the entity in the sentence generated by the LLM as <Type>(such as:"Entity"). We then input the previously generated sentence into the LLM and generate the same sentence with a different entity, then the newly generated entity is stored in E. The fixed prompt used for this scenario is exemplified in the simplified example shown in Figure 1. We present the full prompt on the Figure 3 on Appendix D

Similar to the sentence generation process, we also remove sentences that do not comply with our template and cannot be parsed. This step helps reduce noise introduced by entities added to the entity list.

Methods	BC5CDR	OntoNotes	MIT-R	Tweebank	WNUT-17
Baseline(No augment) #	51 ± 1	58 ± 0	42 ± 1	0 ± 0	0 ± 0
Back-Translation #	53 ± 1	61 ± 1	41 ± 1	3 ± 2	5 ± 1
Pegasus (Zhang et al., 2020) #	52 ± 1	64 ± 0	45 ± 1	3 ± 1	3 ± 1
Paraphrase with DaVinci #	53 ± 2	$\mathbf{67 \pm 0}$	46 ± 2	3 ± 1	4 ± 1
Baseline(No augment)	59 ± 1	58 ± 2	46 ± 1	3 ± 1	3 ± 1
Mixtral Paraphrase	60 ± 1	59 ± 2	46 ± 2	4 ± 1	9 ± 2
Simple Augmentation	57 ± 1	56 ± 2	48 ± 1	4 ± 1	1 ± 0
Only "Generating Sentence"	58 ± 1	58 ± 1	44 ± 1	15 ± 1	16 ± 2
Only "Diversifying Entity"	61 ± 1	61 ± 1	48 ± 1	11 ± 2	6 ± 1
Fixed 5 Human Example	64 ± 1	56 ± 0	49 ± 1	22 ± 2	$\mathbf{23 \pm 3}$
Our Augmentation	$\mathbf{64 \pm 1}$	58 ± 1	$\mathbf{50 \pm 1}$	$\mathbf{23 \pm 2}$	22 ± 1

Table 1: Main result comparing to (Sharma et al., 2023) on 5 dataset. # means that we take the result directly from the papers, the results are rounded so that it can be compared with the baseline paraphrasing results.

3 Experiment and Result

We utilize distill-bert-cased (68M parameters) (Devlin et al., 2018; Sanh et al., 2019) [1] , which employs transformer layers (Vaswani et al., 2017), as our backbone model. Since BERT model representation is token-wise, we pool the representation by summing all tokens belonging to one word, ensuring the output has the same length as the label. A linear head is added to classify each word in a sentence. We train the model using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 2e-5. Training consists of 300 iterations, with each iteration randomly sampling 32 data points from the dataset. The maximum number of entities sampled N is set to 9. The evaluation score used is the micro-F1 score, unless specified otherwise.

To enhance the reproducibility and wider usability of our work, we employ the 4-bit version of Mixtral-8x7B-Instruct-v0.1 (46.7B parameters)[2] as the LLM and set the temperature to 1.

3.1 Main Results

To validate the performance of our proposed method, we conducted experiments on five datasets: BC5CDR (Wei et al., 2016), OntoNotes (Wei et al., 2016), MIT-R (Liu et al., 2013), Tweebank (Jiang et al., 2022), sourced from the TNER project (Ushio and Camacho-Collados, 2022) and WNUT-17 (Derczynski et al., 2017) from the Hugging Face datasets (Lhoest et al., 2021) platform. Our augmentation method is compared against a paraphrase approach and simple augmentation (Dai and Adel, 2020). Results for the five datasets with 1% labeled data and twice the augmentations are presented in Table 1.

Our baseline results show minor discrepancies compared to (Sharma et al., 2023). However, our data generation approach significantly improves the F1 scores on three low-resource datasets: BC5CDR, Tweebank, and WNUT-17, with improvements of 5, 19, and 20 points, respectively. On MIT-R, our improvement is comparable to (Sharma et al., 2023), with an increase of 4 F1 scores.

We attempted to use Mixtral for paraphrasing, and while the results did not show significant improvement, they also did not lead to degradation. Another baseline we explored involved simple augmentation using heuristics (Dai and Adel, 2020), revealing that this method could negatively impact performance on low-resource datasets.

As seen in Table 1, using only the "Generating Sentence" method led to significant improvements on noisy social media datasets such as Tweebank and WNUT-17. This method effectively reduces noise and generates cleaner sentences, contributing more to performance enhancement than solely relying on "Diversifying Entity" in this context.

When using only the "Diversifying Entity" approach, we observed notable improvements in BC5CDR and MIT-R, as well as a significant improvement in OntoNotes, as shown in Table 1. These datasets share the characteristic of having clean sentences, where the primary benefit comes from diversifying the entity pool.

When using only the "Diversifying Entity" approach, we observed notable improvements in

[1] https://huggingface.co/distilbert-base-cased

[2] https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1

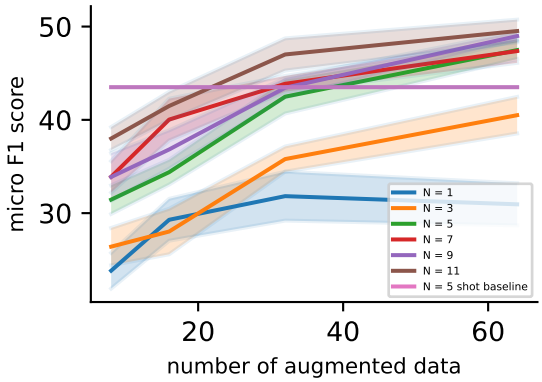

Figure 2: Ablation of the number of entity sampled(N) using 5 seed data on CoNLL-2003

BC5CDR and MIT-R, as well as a significant improvement in OntoNotes. These datasets share the characteristic of having clean sentences, where the primary benefit comes from diversifying the entity pool.

Our augmentation method, which combines both the "Generating Sentence" and "Diversifying Entity" methods, achieves the best results across all low-resource datasets (BC5CDR, MIT-R, Tweebank, and WNUT-17). This demonstrates the complementary nature of the two methods and highlights the importance of addressing both sentence quality and entity diversity for optimal performance gains in low-resource NER.

Interestingly, our data generation method does not enhance performance on the OntoNotes dataset. This discrepancy arises from the dataset's highly imbalanced class distribution, which impacts the effectiveness of our method that uniformly samples classes. Further discussion on this imbalance issue is provided in Section 3.3. Additionally, given OntoNotes dataset size is an order of magnitude larger than the others, and our focus on low-resource scenarios, we primarily emphasize results from the other datasets.

We also observe the robustness of our method to the selection of 5-shot examples. The results using randomly sampled data and fixed human examples are similar. T-tests on the five datasets show statistically insignificant differences, with the numerical results shown in Table 8 in the Appendix. These results suggest that our method is robust to prompt selection; thus, for simplicity, we default to using randomly sampled data.

3.2 Optimal Number of Maximum Entities Sampled on CoNLL-2003

We selected the CoNLL-2003 dataset (Sang and De Meulder, 2003) due to its widespread use in NER research. To investigate how the number of entities sampled during data generation affects model performance, we randomly selected 5 labeled data points from the dataset as the seed.

The impact of varying the number of sampled entities is illustrated in Figure 2. We created five different augmentations for each maximum entity sampled $N \in \{1, 3, 5, 7, 9, 11\}$. For each augmentation, we averaged the results over five random seeds of model evaluation and reported the mean and standard deviation. Figure 2 shows that increasing N generally improves performance, and while there is notable variation between augmentations, this variation reduces with more data generation.

We also compared our data generation with the outcomes obtained using Mixtral for labeling NER with 5-shot learning. Specifically, we selected 5 data points from the CoNLL-2003 train set and utilized them as prompts for Mixtral, following a format similar to that shown in Figure 1. We opted for greedy decoding instead of sampling to achieve optimal performance for the 5-shot Mixtral baseline, resulting in a standard deviation of 0.

In Figure 2, it is clear that as the number of augmentation increase, our proposed method yields higher results compared to the 5-shot Mixtral baseline. Additionally, it is worth noting that directly using 5-shot Mixtral to obtain NER labels presents several challenges, such as the need for prediction alignment. Even when employing 5-shot examples and prompts, Mixtral occasionally outputs extraneous information, making the direct implementation of few-shot tasks less straightforward.

Based on our findings, we selected 9 entities as the default choice, as it demonstrated great performance with lowest variance as the number of augmentation increases. The full numerical result are shown on Table 10 on Appendix

3.3 Application on the Class-Imbalanced Dataset

In this section, we explore the application of our method on class-imbalanced datasets, specifically using the CrossNER (Liu et al., 2021) and 1% labeled data of OntoNotes datasets. While the majority of work utilizes micro-F1 as the evaluation metric, it may not represent all classes equally well.

Methods	Politics	Science	Music	Literat.	AI	Average
No augment	**65.2**/42.7	60.9/39.6	62.9/40.3	51.8/34.7	47.9/33.0	57.8/38.1
Class Uniform	63.6/**44.7**	**61.9/44.5**	**63.7/46.8**	53.3/**38.0**	50.9/**40.0**	**58.7/42.8**
Entity Uniform	63.5/41.6	59.2/38.6	62.9/40.0	**53.6**/37.7	**51.8**/37.2	58.2/39.0

Table 2: CrossNER results, we report micro-F1 and macro-F1 score as (micro/macro)

Method	micro-F1	macro-F1
No augment	57.8 ± 0.4	26.8 ± 0.5
Class Uniform	58.3 ± 0.6	$\mathbf{34.4 \pm 0.3}$
Entity Uniform	$\mathbf{61.0 \pm 0.9}$	31.3 ± 1.7

Table 3: Different sampling 1% of the OntoNotes dataset with twice the augmentation of gold data.

Some datasets have classes that constitute less than 1% of all data. Therefore, in this section, we also evaluate the macro-F1 score.

The use of LLM data generation allows us to address the problem by controlling the probability of sampled classes. In this section, we compare two types of entity type sampling. The first method uniformly samples classes, referred to as "class uniform," while the second samples proportionally based on the number of entities in each entity type, called "entity uniform."

The results for OntoNotes are presented in Table 3. It is observed that the entity uniform method achieves the highest micro-F1, while the class uniform method attains the best macro-F1. The results on the class-imbalanced low-resource CrossNER dataset demonstrate a significant benefit from using the class uniform method. In contrast to OntoNotes, the advantage of entity uniform is only observed in literature and AI datasets from CrossNER and shows only marginal improvement compared to class uniform. We attribute this difference to the larger number of data points in OntoNotes, resulting in a higher volume of augmented examples.

3.4 CheckList Evaluation

Robustness evaluation of NLP models is crucial for assessing their reliability in real-world applications. In this study, we employ the CheckList framework (Ribeiro et al., 2020) to evaluate the performance of our model trained on the CoNLL-2003 dataset using the 5-shot setting and focuses on the robustness against perturbations in person entity names. We augment data with quantities ranging from 0 to 64, with increments of 8, and the results are averaged over 5 seeds.

Method	English	Vietnamese	Brazilian
Spacy	4.3%	31.0%	25.7%
0 Aug	14.9%	71.5%	88.0%
8 Aug	3.9%	60.5%	77.1%
16 Aug	0.1%	29.7%	50.9%
32 Aug	0%	19.4%	23.5%
64 Aug	0%	8.2%	9.3%

Table 4: Results of the CheckList Invariance test error rate. Spacy refer to the Spacy en_core_web_sm model.

For assessing the model's robustness regarding person entity names, we conduct an Invariance test (INV) based on the provided code[3]. This test involves manipulating person entity names within a standardized sentence template and examining the consistency of the model's predictions. Specifically, we utilize the template "I met with <firstName> <lastName> last night," where the names are substituted with English, Vietnamese, and Brazilian variants. Each language variation comprises 300 examples for thorough evaluation.

Our findings, as presented in Table 4, indicate a positive correlation between augmented entity data and model robustness against name perturbations. Notably, our model exhibits superior performance compared to the Spacy (Honnibal et al., 2020) baseline, particularly under augmented data conditions equal to or exceeding 32.

4 Conclusion

We propose a simple method for generating new data for NER tasks using only few examples. Our approach is simple and leverages open-source LLM, making it more affordable and cost-effective to reproduce. The method demonstrates performance improvement across various datasets and topics. Additionally, we showcase the application of our NER data generation methods on class-imbalanced datasets, highlighting significant performance improvements on such datasets.

[3]https://github.com/marcotcr/checklist

Limitations

Our proposed methods rely on LLM, demanding substantial resources, even with the 4-bit quantized version. Utilizing the 4-bit Mixtral requires approximately 34GB of GPU VRAM. Furthermore, our method is time-intensive, requiring over 10 seconds to generate one data using 2 x 24GB RTX 3090 GPUs. Moreover, our method may introduce noise, particularly with larger datasets.

As our study only focus on single LLM for our experiments, namely 4-bit version of Mixtral-8x7B-Instruct-v0.1. We acknowledge that our findings may not generalize to weaker models and that the performance and robustness of our proposed methods could be influenced by the specific characteristics and capabilities of the chosen LLM. Additionally, there is a potential risk of data leakage if Mixtral has been trained on the dataset used for evaluation in this paper. Despite Mixtral being open-source, the details of its training data remain undisclosed.

While our study focuses solely on English datasets, it is important to note that the results may not directly transfer to other languages. This limitation arises from the possibility that LLMs in other languages may not possess the same level of proficiency as their English counterparts. Nonetheless, we anticipate that advancements in multilingual LLM performance will eventually facilitate the extension of our findings to other languages.

Ethics Statement

As our method generates data based on randomly selected entities, there is a higher likelihood of producing factually incorrect sentences that potentially introduce bias into the model.

References

Partha Sarathy Banerjee, Baisakhi Chakraborty, Deepak Tripathi, Hardik Gupta, and Sourabh S Kumar. 2019. A information retrieval based on question and answering and ner for unstructured information without using sql. *Wireless Personal Communications*, 108:1909–1931.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

John Joon Young Chung, Ece Kamar, and Saleema Amershi. 2023. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. *arXiv preprint arXiv:2306.04140*.

Xiang Dai and Heike Adel. 2020. An analysis of simple data augmentation for named entity recognition. *arXiv preprint arXiv:2010.11683*.

Leon Derczynski, Eric Nichols, Marieke Van Erp, and Nut Limsopatham. 2017. Results of the wnut2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jinyuan Fang, Xiaobin Wang, Zaiqiao Meng, Pengjun Xie, Fei Huang, and Yong Jiang. 2023. Manner: A variational memory-augmented model for cross domain few-shot named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4261–4276.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*.

Ruohao Guo and Dan Roth. 2021. Constrained labeled data generation for low-resource named entity recognition. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4519–4533.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Hang Jiang, Yining Hua, Doug Beeferman, and Deb Roy. 2022. Annotating the tweebank corpus on named entity recognition and building nlp models for social media analysis. *arXiv preprint arXiv:2201.07281*.

Hyunjae Kim, Jaehyo Yoo, Seunghyun Yoon, and Jaewoo Kang. 2022. Automatic creation of named entity recognition datasets by querying phrase representations. *arXiv preprint arXiv:2210.07586*.

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

Phong Le and Ivan Titov. 2018. Improving entity linking by modeling latent relations between mentions. *arXiv preprint arXiv:1804.10637*.

Q Lhoest, AV del Moral, Y Jernite, A Thakur, P von Platen, S Patil, J Chaumond, M Drame, J Plu, L Tunstall, et al. 2021. Datasets: a community library for natural language processing. arxiv. *arXiv preprint arXiv:2109.02846*.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. Bond: Bert-assisted open-domain named entity recognition with distant supervision. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1054–1064.

Jian Liu, Yufeng Chen, and Jinan Xu. 2022. Low-resource ner by data augmentation with prompting. In *IJCAI*, pages 4252–4258.

Jingjing Liu, Panupong Pasupat, Yining Wang, Scott Cyphers, and Jim Glass. 2013. Query understanding enhanced by hierarchical parsing structures. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 72–77. IEEE.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13452–13460.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Jonathan Raiman and John Miller. 2017. Globally normalized reader. *arXiv preprint arXiv:1709.02828*.

Xiang Ren, Ahmed El-Kishky, Chi Wang, Fangbo Tao, Clare R Voss, and Jiawei Han. 2015. Clustype: Effective entity recognition and typing by relation phrase-based clustering. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 995–1004.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Timo Schick, Jane Dwivedi-Yu, R Dessı, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools (2023). *arXiv preprint arXiv:2302.04761*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2024. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36.

Saket Sharma, Aviral Joshi, Yiyun Zhao, Namrata Mukhija, Hanoz Bhathena, Prateek Singh, and Sashank Santhanam. 2023. When and how to paraphrase for named entity recognition? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7052–7087.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. 2023. Does synthetic data generation of llms help clinical text mining? *arXiv preprint arXiv:2303.04360*.

Asahi Ushio and Jose Camacho-Collados. 2022. T-ner: an all-round python library for transformer-based named entity recognition. *arXiv preprint arXiv:2209.12616*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Somin Wadhwa, Silvio Amir, and Byron C Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2023, page 15566. NIH Public Access.

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. *arXiv preprint arXiv:2010.03851*.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2020. Automated concatenation of embeddings for structured prediction. *arXiv preprint arXiv:2010.05006*.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2016. Assessing the state of the art in biomedical relation extraction: overview of the biocreative v chemical-disease relation (cdr) task. *Database*, 2016.

Usama Yaseen and Stefan Langer. 2021. Data augmentation for low-resource named entity recognition using backtranslation. *arXiv preprint arXiv:2108.11703*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International conference on machine learning*, pages 11328–11339. PMLR.

Ran Zhou, Xin Li, Lidong Bing, Erik Cambria, and Chunyan Miao. 2023. Improving self-training for cross-lingual named entity recognition with contrastive and prototype learning. *arXiv preprint arXiv:2305.13628*.

Ran Zhou, Xin Li, Ruidan He, Lidong Bing, Erik Cambria, Luo Si, and Chunyan Miao. 2021. Melm: Data augmentation with masked entity language modeling for low-resource ner. *arXiv preprint arXiv:2108.13655*.

Dawei Zhu, Xiaoyu Shen, Marius Mosbach, Andreas Stephan, and Dietrich Klakow. 2023. Weaker than you think: A critical look atweakly supervised learning. *arXiv preprint arXiv:2305.17442*.

A Dataset Statistic

In this section, we present the primary dataset utilized in our main experiment, as well as the augmented dataset generated for experimental augmentation purposes.

A.1 Benchmark Dataset

We offer comprehensive insights into the datasets employed throughout our experiments. Specifically, we outline the datasets utilized, including the respective classes within each dataset. Table 5 presents detailed statistics regarding the number of data instances for each dataset.

BC5CDR: The BioCreative V CDR NER dataset focused in the biomedical domain. This dataset comprises two named entity types of chemical and disease.

OntoNotes: The OntoNotes 5 dataset is a NER dataset in the News domain. It encompasses 18 named entity types, including cardinal, date, event, facility, geopolitical entity, language, law, location, money, affiliation, ordinal, organization, percent, person, product, quantity, time, and work of art.

	Train	Valid	Test
TNER			
BC5CDR	5228	5330	5865
OntoNotes	59924	8528	8262
MIT-R	6900	760	1521
Tweebank	1639	710	1201
HF datasets			
WNUT-17	3394	1009	1287
CoNLL-2003	14041	3250	3453
Cross NER			
Politics	200	541	651
Science	200	450	543
Music	100	380	456
Literature	100	400	416
AI	100	350	431

Table 5: Dataset statistic

MIT-R: The MIT Restaurant dataset is a NER dataset in the Restaurant domain. It includes 8 named entity types of rating, amenity, location, restaurant name, price, hours, dish, and cuisine.

TweeBank NER: The TweeBank NER dataset is in the Twitter domain and encompasses 4 named entity types: location, person, organization, and miscellaneous.

WNUT 2017: The WNUT 2017 dataset is a NER shared task focused on detecting entities with unusual surface forms or those that are rare. This dataset is also in the Twitter domain and includes 6 named entity types of corporation, creative work, group, location, person, and product.

CoNLL-2003: The CoNLL-2003 dataset is a NER shared task in the news domain designed for language-independent NER. It consists of 4 named entity types of persons, locations, organizations, and names of miscellaneous entities.

CrossNER: The CrossNER dataset is a NER dataset collected from Wikipedia, featuring five diverse domains with specialized entity categories for each domain. The dataset includes diverse entity types, limited data, and a relatively unbalanced distribution of entities.

Politics: The Politics subset of CrossNER consists of 10 entities, including politician, person, organization, political party, event, election, country, location, and miscellaneous.

Science: The Natural Science subset of CrossNER comprises 17 entities, such as scientist, person, university, organization, country, location, discipline, enzyme, protein, chemical compound,

chemical element, event, astronomical object, academic journal, award, theory, and miscellaneous.

Music: The Music subset of CrossNER includes 13 entities, encompassing music genre, song, band, album, musical artist, musical instrument, award, event, country, location, organization, person, and miscellaneous.

Literature: The Literature subset of CrossNER consists of 11 entities, including book, writer, award, poem, event, magazine, person, location, organization, country, and miscellaneous.

AI: The Artificial Intelligence subset of CrossNER contains 12 entities, such as field, task, product, algorithm, researcher, metrics, university, country, person, organization, location, and miscellaneous.

A.2 Augmented Data Statistic

We present the statistics of our generated dataset in Table 6. In this table, the 'Real' column signifies the number of labeled data points used in the main experiment for each dataset, 'Aug' indicates the number of generated data points included for each dataset, and '+Entity' denotes the count of unique entities added to the dataset.

For the CoNLL-2003 dataset, we conducted five data generation processes, each generating 64 data points. The table presents the average and standard deviation of unique entities added across these five runs. During each run, a maximum of 9 entities were sampled during the sentence construction process. The CrossNer section provides a breakdown of added entities for both class uniform (CU) and entity uniform (EU) data generation scenarios.

Table 6 demonstrates the effectiveness of our data generation method in creating diverse amounts of novel entities across different datasets and generation scenarios.

	Real	Aug	+Entity
Main Result			
BC5CDR	52	104	113
OntoNotes	599	1198	881
MIT-R	69	138	115
Tweebank	16	32	33
WNUT-17	32	64	53
Entity Ablations			
CoNLL-2003	5	64	97±7
CrossNER(CU/EU)			
Politics	200	400	513/486
Science	200	400	558/488
Music	100	200	280/235
Literature	100	200	267/277
AI	100	200	220/260

Table 6: Augmented Dataset statistic

B Additional Related Works

Data augmentation is a versatile technique used to improve the performance of NLP models by increasing the amount and diversity of training data (Feng et al., 2021). Its model-agnostic nature allows for easy integration with various NLP tasks. In this section, we explore the growing trend of using LLMs for data generation. Subsequently, we will narrow our focus to data generation in NER task.

B.1 Data Generation Using LLM

The increasing capabilities of LLMs have led to a surge in their use for data generation, achieving remarkable success across diverse NLP tasks. For instance, LLMs have demonstrated the ability to generate data that significantly improves classification accuracy (Chung et al., 2023), enhances relation extraction performance (Wadhwa et al., 2023), and even enables self-learning of tool usage (Schick et al., 2024). These applications highlight the potential of LLMs as a powerful tool for creating high-quality training data, especially for smaller models with limited data resources.

B.2 Data Generation for NER

Several existing works have explored the use of neural language models for data generation in NER. Here, we compare our approach with these methods, focusing on key aspects such as reliance on external data, ability to work in few-shot settings, and the diversity of generated entities and sentences. Table 7 summarizes the comparison and highlights the unique strengths of our proposed method.

label-conditioned word replacement(LCWR) Label-conditioned word replacement (LCWR) methods, as explored in (Zhou et al., 2021) and (Liu et al., 2022), often require substantial amounts of data for training, making them less practical for few-shot scenarios and introducing additional complexity. Moreover, their focus on simply replacing existing named entities restricts their ability to generate diverse sentence structures and novel entity types, potentially leading to limited performance

Criterion	LCWR	CD	GPT 3 Paraphrase	(Tang et al.)	Ours
Work without external data	O	X	O	O	O
Work without training	X	X	O	O	O
Work on few-shot data	X	X	O	O	O
Use open model	O	O	X	X	O
Diverse entity	O	O	X	X	O
Diverse sentence	X	O	O	O	O

Table 7: Comparison of our proposed method with prior works (LCWR: Label-Conditioned Word Replacement, CD: Constrained Decoding) based on key criteria for data generation in low-resource NER tasks.

	5 Human Example	Random Sample	t-statistic
BC5CDR	64.802 ± 0.991	63.788 ± 0.623	1.937
OntoNotes	56.535 ± 0.790	56.464 ± 0.672	0.1531
MIT-R	49.174 ± 1.451	50.074 ± 0.844	1.1989
Tweebank	22.759 ± 1.763	22.789 ± 1.574	0.0284
WNUT-17	23.692 ± 2.662	21.884 ± 1.218	1.3810

Table 8: T-test results on TNER

improvements. While (Liu et al., 2022) proposes a 0-shot LCWR approach, it relies on external data sources, a constraint our proposed method avoids by directly leveraging the capabilities of LLMs for generating both entities and sentences.

constrained decoding(CD) Constrained decoding (CD) methods, while effective for low-resource NER, often operate under the assumption of a readily available source dataset and require training to adapt to the target dataset. This is exemplified in (Guo and Roth, 2021), where CD leverages resources from a high-resource language to improve NER in a low-resource setting. In contrast, our proposed method using LLMs for data generation does not rely on a source dataset or require additional training, making it more versatile and applicable to scenarios where such resources are scarce. However, the potential for combination remains, as the data generated by our LLM-based approach can be subsequently utilized in conjunction with CD techniques for further refinement and adaptation, potentially leading to enhanced performance in low-resource NER tasks.

GPT 3 Paraphrase As Table 1 illustrates, paraphrasing methods like GPT-3 Paraphrase (Sharma et al., 2023) and backtranslation (Yaseen and Langer, 2021) offer limited benefits for data augmentation in low-resource NER. The scarcity of entities in such datasets restricts the ability to generate diverse and meaningful paraphrases. Our proposed method, in contrast, tackles this challenge by utilizing LLMs to generate both novel entities and sentences, thus expanding the potential for data augmentation and achieving more significant performance gains in low-resource scenarios.

(Tang et al.) The work by (Tang et al., 2023) presents the most similar approach to ours. However, key differences exist. We employ a "Diversifying Entity" technique, where the LLM is used to generate new entities based on existing examples, ensuring a wider range of entities for data generation. This contrasts with the assumption made by Tang et al. that a large entity list is already available. Additionally, our method samples multiple entities for each generated sentence, leading to more diverse and complex examples compared to their single-entity approach. The impact of these differences is evident in the improved performance observed in Table 1 and Figure 2.

C Numerical Results

This section provides further analysis to support and expand upon the findings presented in the main paper. We first present the results of T-tests conducted in Section 3.1. Next, we compare the performance of our proposed augmentation method against a baseline using the Mixtral model with 5-shot learning to evaluate its effectiveness. Finally, we conduct an ablation study to understand the individual contributions of different components within our proposed augmentation method.

Method	BC5CDR	OntoNotes	MIT-R	Tweebank	WNUT-17
Baseline without any augmentation	58.51	58.19	46.45	3.40	2.56
Only "2.1 Generating Sentence"	58.73	58.33	44.21	14.86	15.91
Only "2.2 Diversifying Entity"	61.21	**60.88**	48.31	11.81	6.29
Our Augmentation (Both method)	**63.79**	56.46	**50.07**	**22.79**	**21.88**

Table 9: Ablation study

Method	micro-F1 score
1 Entity	30.946 ± 4.505
3 Entity	40.499 ± 3.922
5 Entity	47.478 ± 1.904
7 Entity	47.372 ± 2.522
9 Entity	**48.981 ± 0.875**
11 Entity	49.522 ± 2.438
5-shot Mixtral	43.503 ± 0.0

Table 10: Results of the 5-shot Mixtral model using 5 data points to create 64 augmented data on CoNLL-2003 with different maximum entity samples (N).

C.1 T-test Result

We provide an in-depth exploration of the T-test methodology employed in Section 3.1. The corresponding t-statistics are detailed in Table 8. In this table, we compare our method that utilizes human examples with one that randomly samples from the dataset. The mean and standard deviation are computed by averaging results from 5 experiments. Notably, all computed t-statistics fall below 2 (95% confidence interval), leading us to conclude that no significant differences exist. This finding suggests that our proposed method is robust and not overly sensitive to the specific choice of examples used for few-shot learning. This robustness further strengthens the practicality and generalizability of our approach for data augmentation in low-resource NER tasks.

C.2 5-shot Mixtral baseline on CoNLL-2003

In this section, we also present the numerical results from Figure 2. As we can see in Table 10, using 9 sampled entities when creating sentences results in similar performance to using 11 entities, but it has far less variance. Thus, we use 9 entities as the default.

We also show the result of using 5-shot Mixtral, which demonstrates that our proposed augmentation can significantly outperform it when using 5 or more sampled entities with 64 augmented data.

We did not investigate the 0-shot performance of Mixtral for several reasons. Firstly, our data generation procedure also uses 5 data points from the CoNLL 2003 dataset. Therefore, to ensure better comparability, we chose to evaluate the 5-shot performance. Additionally, making 0-shot predictions for NER to adhere to a fixed structure is extremely difficult, making evaluation challenging.

D Full Prompt

In the main paper, the full prompt for all used prompts is shown in Figure 3. The Mixtral-8x7B-Instruct-v0.1 model treats the prompt as a user and assistant interaction. We use instruction by forcing the model to respond with 'Ok'. We present few-shot examples as user-assistant interactions and obtain query results by appending the user's message to the prompt and feeding it to the LLM. Inline instructions are included after each user prompt to help mitigate noise introduced by Mixtral LLM's

Generating Sentence Prompt: We provide the full prompt of the fixed 5 human examples used in the main paper in Figure 3. For the random 5-shot, we replace the 5 fixed human examples with randomly sampled data from the datasets, adding the tags and instructions.

Paraphrasing Prompt : In Figure 3, the paraphrasing prompt is the one used to create Mixtral paraphrases in Table 1. We randomly sample data from the dataset and generate its paraphrase using the displayed fixed prompt.

Diversifying Entity Prompt: In Figure 3, the diversifying entity prompt is the one used to create new entities that will be added to the entity list E. We feed the sentence generated by LLM using the generating sentence prompt, change the format, and directly feed it again to the diversifying entity prompt.

E Generated Data Example

We provide examples of data created by our method from Figure 4 to 14. The example shows that the proposed method able to create diverse data and adheres to the data distribution of real datasets.

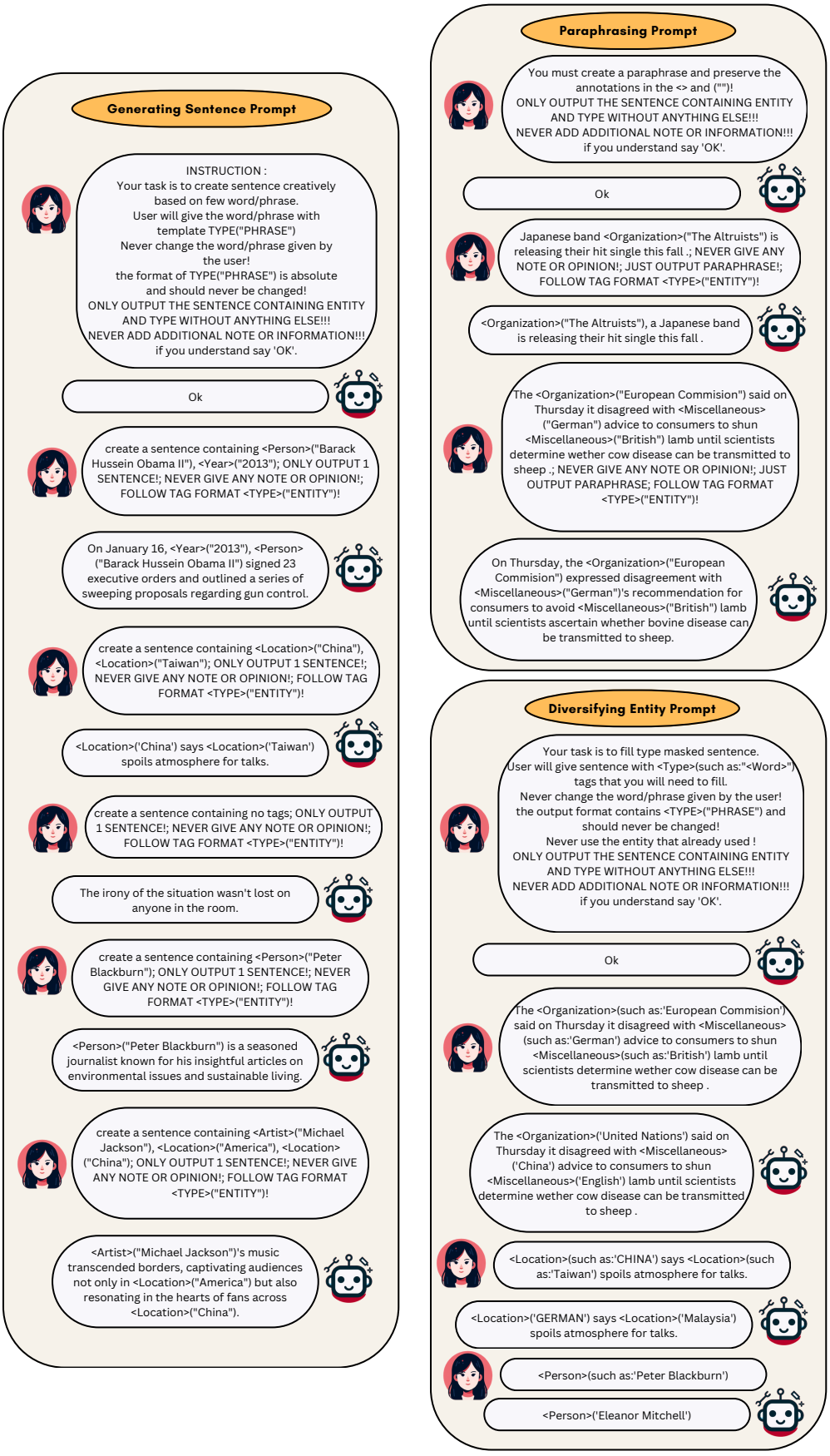

Figure 3: full prompt

CoNLL-2003

Example 1 :

In the NETHERLANDS Location based AMA Organization film festival , jury members Carlotta Chen Person from Greek Location and MAX SILVERMAN Person from Japanese Location along with festival director Minako Mori Person are preparing for the screening of various films .

Example 2 :

The Canadian Miscellaneous PM and the World Bank Organization both welcomed the French Miscellaneous president's remarks that British Miscellaneous and Singapore Location cooperation in the Indo-Pacific region was a priority .

Figure 4: generated CoNLL-2003 data examples

WNUT 17

Example 1 :

John Mayer Person gets new role with Bose Corporation and gives Beatles' fanclub Group all his NBA Group gear http : / / bzfd . it / 1jY6GYF John Mayer Person on the new role w / Bose Corporation & amp ; his Beatles' fanclub

Example 2 :

Company Brochure CreativeWork details all the benefits of the Membership Plan CreativeWork in UK Location - Check it out here : [www . getourbrochure . co . uk Location] (http : / / www . getourbrochure . co . uk Location) !

Figure 5: generated WNUT-17 data examples

Tweebank

Example 1 :

RT @ USER2478 : @ Silvio Berlusconi Person 's North American Tour 2011 Miscellaneous event in Nigeria Location was so hype , he got exclusive access to Facebook Organization features ! :)

Example 2 :

Before going on the North American Tour 2011 Miscellaneous , Jack Person performed in Iran Location at a festival . Amaaaaazing ! @ USER568

Figure 6: generated tweebank data examples

BC5CDR

Example 1 :

After receiving isoflurane Chemical during electroconvulsive therapy for major depression , a patient with multiple sclerosis Disease showed significantly reduced airway responsiveness to theophylline Chemical .

Example 2 :

In this study , the effects of benzodiazepines Chemical on the incidence of extrapyramidal signs Disease after intrastriatal administration of puromycin aminonucleoside Chemical were examined in mice treated with hydrogen peroxide Chemical to induce oxidative stress .

Figure 7: generated BC5CDR data examples

MIT Restaurant

Example 1 :

has the in n out burger RestaurantName closed at 8 pm TimeHours ?

Example 2 :

i want to try some american CuisineType , chinese buffet CuisineType meals with delightful RatingType fries CuisineType next time .

Figure 8: generated MIT restaurant data examples

Ontonotes

Example 1 :

Make sure to arrive before 7 : 15 PM Time if you want to get a good seat .

Example 2 :

Vatican GeoPoliticalEntity archbishops do not endorse trucks Product adorned with Picasso's 'Guernica' WorkOfArt , and Navy Organization ships do not keep Bank of New York Co . Organization money on board them .

Figure 9: generated Ontonotes data examples

AI

Example 1 :

Grace Yang Person utilized Python language ProgrammingLanguage and R ProgrammingLanguage while working on the sign language Task translation project in Paris Location , evaluating the performance of the line search algorithm Algorithm with the help of the F1 score Metrics .

Example 2 :

Recipient of a S . V . della laurea ad honorem Miscellaneous degree from the Cognitive Science Society Organization , he is a consultant and board member with Paramount Organization after 20 years in the industry .

Figure 10: generated AI data examples

Music

Example 1 :

To promote Artists Against Discrimination Organization and Equal Rights Association Organization , Egypt Location hosted a Summer Festival Event that included the concert of Lyre MusicalInstrument playing artists from ITALY Country , winner of the Grammy Award for Best Recording Package Award for their album released that year .

Example 2 :

My favorite show Miscellaneous on Lion King Miscellaneous was at the The Closing Ceremony Event .

Figure 11: generated music data examples

Science

Example 1 :

Craig Mello Scientist , from Brazil Country , won the Copley Medal of the Royal Society Award in 2008 for his research on Dicer Enzyme , a key component for RNA Miscellaneous interference (RNAi) and its role in genetic regulation , as well as research on the development of neurons (Neuron Miscellaneous) .

Example 2 :

Apart from researching the KRT8 Protein , the Nobel Prize Award awarded scientist in chemistry Discipline and biochemistry Discipline was also closely following the Stock Exchange Market Reform Event .

Figure 12: generated science data examples

Literature

Example 1 :

From the Foundation series Miscellaneous to The Trumpet Player Poem and Leonardo da Vinci Person 's sketches , artists across time Magazine and genres have drawn inspirationfrom the beauty of music and its instruments .

Example 2 :

Edgar Allan Poe Award Award for Best Novel Literarygenre honored works such as Ağır Roman Book , Records of the Grand Historian Book , The Tempest Poem and Around the World in Eighty Days Book .

Figure 13: generated literature data examples

Politics

Example 1 :

Ezekiel Politician , a Sudanese national , was arrested in Hitchin Location after his identity was discovered using forged Sudanese Air Force Miscellaneous documents .

Example 2 :

During the War of Liberation Event in Iran Country , Porfirio Lobo Sosa Politician 's Peasant Parties PoliticalParty received support from people in Honduras Country and Kyaukse Location as well as Sonia Singh Person , a famous TV journalist .

Figure 14: generated politics data examples