Augmenting and Refining NER Datasets through LLMs

Anonymous ACL submission

Abstract

Traditional named entity recognition (NER) dataset annotation methods often suffer from high costs and inconsistent quality. This study proposes a hybrid annotation approach that combines human effort with large language models (LLMs) to reduce noise due to the missing labels and improve NER model performance cost-effectively. A label-mixing strategy is also introduced to address class imbalance. Our experiments demonstrate that the proposed method outperforms traditional approaches, even under limited budget constraints.

1 Introduction

Named entity recognition (NER) is a critical task that involves identifying and classifying named entities in a given text into predefined categories (Mohit, 2014). NER models' performance relies heavily on the quality of annotations in the datasets (Mitkov et al., 2000). Currently, most datasets employed for training NER models are annotated by humans. This manual annotation process is fraught with several challenges, such as annotation errors, time and financial costs (Katiyar and Cardie, 2018; Fort et al., 2011).

This study introduces a novel method to address the challenges of manual annotation by leveraging large language models (LLMs) as annotators. The approach aims to improve dataset quality, which enhances the performance of NER models. Using LLMs for annotation offers several advantages. Human annotators can be subjective and suffer from fatigue, which may reduce accuracy. In contrast, LLMs offer consistency and efficiency throughout the annotation process. Furthermore, while employing LLMs as predictors leads to increasing costs with repeated predictions, employing them as annotators only requires a one-time cost during the annotation process. Afterward, training a predictor based on small models, such as BERT, with the improved dataset can provide adequate performance for the NER task.

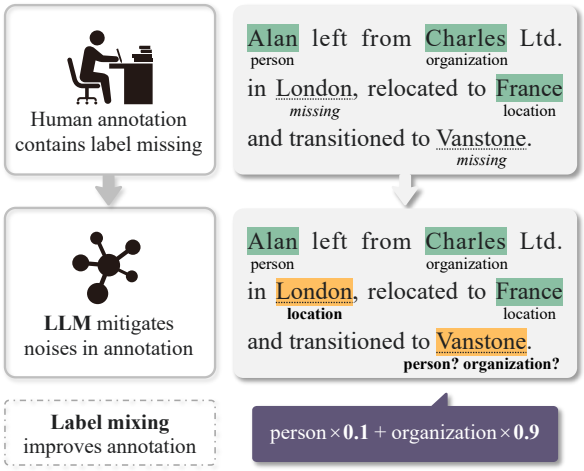

Figure 1: LLM-based annotation and label mixing.

Our approach using LLMs as annotators reduces the noise of missed labels by integrating manual and LLM-based annotations (as shown in Figure 1), thus improving the quality of the data set for NER model training in a cost-effective and automatic manner. We find that LLMs tended to assign multiple named entity labels to a single expression. Additionally, an imbalance in data volume across labels leads to performance disparities among labels. To address this, we introduce a label mixing technique that integrates multiple named entity labels, reducing the variance of performance among labels and enhancing the model's robustness.

Comprehensive evaluation experiments validated the effectiveness of our approach, demonstrating its ability to train NER models on noisy datasets, such as those with missed labels, while requiring minimal financial cost.

2 Related Work

In machine learning, the accuracy and reliability of the dataset used for model training are critical factors directly influencing the performance of the

trained model. In this context, research has been conducted to scrutinize the reliability of datasets used for training NER models. Research, such as the one presented in Han et al. (2023a), highlights the inherent errors in these datasets. These inaccuracies range from mislabeled entities to inconsistent categorization, impacting the overall performance and reliability of NER models (Mishra et al., 2020; Mitkov et al., 2000).

There are several challenges in automatic annotation. Ghosh et al. (2023) address the data imbalance problem through an attention map-based data expansion. Shen et al. (2023b) utilize multiple prompt templates to address the problem of multiple label assignments to the same entity. Shen et al. (2023a) and Zhang et al. (2022) address the issues of noise and generative bias by utilizing causality-based methods. To address these problems, we propose a hybrid approach that combines manual and automatic LLM annotations.

3 Methodology

3.1 LLM-based Annotation

In the context of automating linguistic data using LLMs, the GPT-NER method, as described by Wang et al. (2023), provides a key approach. This method plays a significant role in our study by leveraging LLMs for automated annotation. A core aspect of GPT-NER is the ability of the LLM to identify and mark specific entities within text, surrounding these with designated character strings, such as *"I live in @@Tokyo##"*.

The selection of few-shot examples is based on the approach outlined by Wang et al. (2023), where examples are chosen from validation data using cosine similarity between token embeddings in the input text.

Additionally, this approach enables the LLM to understand both the inherent properties of entities and their relationships within the input-output framework by incorporating few-shot examples. Based on this methodology, we designed the prompt structure as described above.

In this prompt structure, the [*target entity type*] slot is used for different entity types, such as `PER`, `ORG`, `LOC`, and `MISC`. The [*definition target entity type*] slot is allocated for explicitly defining these entity types. The [*Example K Input*] and [*Example K Output*] slots are designated for the input text and its corresponding output, serving as few-shot examples.

Instructions for the Large Language Model

You are an excellent linguist.
Identify [*target entity type*] entities in given text.
If the text contains no [*target entity type*] entities, replicate the input text without any changes.
Strictly adhere to the definition provided below.
Definition of [*target entity type*]:
[*definition target entity type*]
[*Example* 1 *Input*]
[*Example* 1 *Output*]
[*Example* 2 *Input*]
[*Example* 2 *Output*]
...
[*Example K Input*]
[*Example K Output*]

The entity recognition process is systematically applied to each target entity type in the text for annotation. In cases where multiple iterations are required for different entity types, a single word may receive multiple labels. To handle such overlapping labels, we use a simple approach: one label is randomly selected from the multiple assignments.

The few-shot examples are chosen based on the method described by Wang et al. (2023), which involves selecting examples from validation data based on the cosine similarity between token embeddings in the input text.

3.2 Hybrid: LLM-Based and Manual Annotation

To enhance the quality of the NER dataset, we created hybrid annotations by merging labels from LLMs with manually generated ones. This process involved carefully analyzing both sources to ensure consistency and accuracy. Entity labels from the manual data were prioritized, with LLM-based labels added only when they did not overlap. As manual annotations are considered higher quality, they were preferred over LLM-generated ones. The resulting merged dataset offers a more robust and diverse set of labels, reducing biases and errors from either source.

3.3 Label Mixing in LLM-Based Annotation

Addressing the issue, as mentioned earlier, of multiple labels being assigned to a single word, we extended the Mixup (Zhang et al., 2018) technique, traditionally used in image processing, to NER tasks. Mixup is a data augmentation method that improves model performance and robustness by mixing two inputs and outputs. Our approach involves blending two different labels for a single

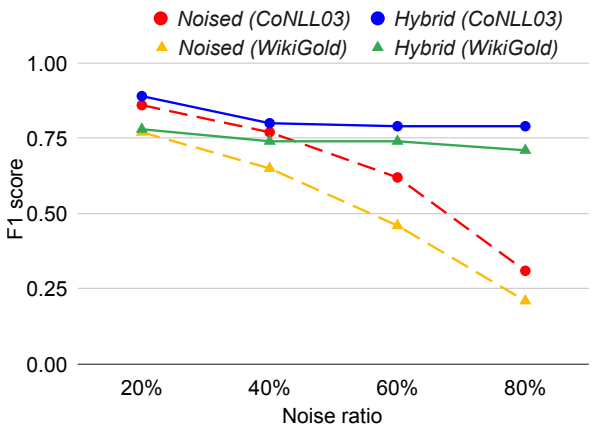

Figure 2: Comparison of model performance when using noised and datasets annotated by Hybrid approach.

Budget	#Manual	#LLM	F1 Score
$38	87	0	0.02
	70	702	0.81
	35	2,106	0.84
	0	3,510	**0.85**
$152	351	0	0.63
	280	2,808	0.85
	140	8,424	**0.87**
	0	14,041	0.86
$608	1,404	0	0.86
	1,263	5,616	**0.87**
	1,123	11,232	**0.87**

Table 1: Comparison between different data count balances from manual and LLM-based data under the fixed budgets.

token to create new labels. This method is formally represented as follows:

$$\boldsymbol{y}_{\text{mix}} = \lambda \times \boldsymbol{y}_i + (1 - \lambda) \times \boldsymbol{y}_j$$

Where $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ are one-hot encodings of two different labels annotated entity types, and $\lambda \in [0, 1]$ is a parameter determining the mixing ratio. Following the mixup technique (Zhang et al., 2018), λ is drawn from the beta distribution ($\alpha = \beta = 0.2$). Compared to randomly selecting one label from multiple labels assigned to a single word, this technique contributes to a more nuanced understanding of tokens that could belong to multiple entity types, enhancing the dataset's richness and classification flexibility.

4 Experiments

Setup. We conducted experiments using two distinct datasets. Firstly, we employed the CoNLL03 (Tjong Kim Sang and De Meulder, 2003), recognized as a standard dataset for NER. Additionally, acknowledging recent advancements in this field that have led to a saturation in performance improvements on the CoNLL03, we also included the WikiGold (Balasuriya et al., 2009). The WikiGold dataset was specifically chosen due to its higher level of challenge and the observation that performance levels achieved on it have not yet paralleled those on the CoNLL03. This was split into training, validation, and test sets in a 7:1:2 ratio for the experiments.

For the automated annotation process, we leveraged ChatGPT (`gpt-3.5-turbo-0613`)[1], and we set the number of few-shot examples at 32.

Finally, to validate the effectiveness of our annotation approach, we fine-tuned a sequence classification model based on a pre-trained BERT (Devlin et al., 2019), utilizing the datasets annotated through this methodology. The evaluation of the NER task is in line with Tjong Kim Sang and De Meulder (2003), and we present weighted average F1 scores in this paper.

A detailed description of the hardware and software configurations is also provided in Appendix A to ensure reproducibility.

Noise Recovery with LLM-based Annotations. We assessed the capability of LLM-based annotations to recover from noisy data. Han et al. (2023b) pointed out some annotation errors, such as missing annotations or label switching, in the CoNLL03 dataset (Tjong Kim Sang and De Meulder, 2003). We prepared noised data that reproduces this problem by removing annotations for some entity. We show that it can be improved by combining LLM-based annotation in the previously described manner. We experimented with 20%, 40%, 60%, 80% entity annotations removed. By comparing the noised datasets to those enhanced through our proposed methodology, we established a baseline for improvement. Figure 2 presents a comparative analysis of these datasets, highlighting the efficiency of our approach to noise reduction. It was shown that even when noise is severe, it can be significantly improved by combining LLM-based annotations.

Comparison Under the Same Budget. We evaluated our approach under identical budget constraints by drawing parallels with the study of Ding et al. (2023). Our experiments were conducted on three different budgets. We created composite datasets including different numbers of samples from manual and LLM-based annotations. The

[1] https://chat.openai.com

Annotations	CoNLL03 mean	CoNLL03 variance	WikiGold mean	WikiGold variance
LLM-based w/o label mixing	0.83	0.011	**0.63**	0.043
LLM-based w/ label mixing	**0.84**	**0.008**	**0.63**	**0.032**
Manual	0.89	0.005	0.80	0.009

Table 2: Comparison on the average F1-score and variance between before and after label mixing.

middle budget ($152) is the cost of LLM-based annotation of all training data of the CoNLL03 dataset, and the costs of a quarter and a quadruple of it are assumed for small ($38) and large ($608) budgets, respectively. The relationship between the balance of the number of data in each budget and performance is shown in Table 1.

When the budget is small ($38), training is not possible with only manual data due to the small amount of data. Table 1 shows that it is thought that performance can be improved by using LLM-based annotations to increase the training data. For the middle budget ($152), the model's performance was reduced using only the manual data; using just 2,808 LLM-based annotations showed a significant improvement. In addition, the performance of the Hybrid annotations was higher than that of LLM-based alone, suggesting that the use of both annotations contributes to the performance. For the large budget ($608), the performance is sufficient with 1,404 manual data, but a slight performance improvement was confirmed by reducing the manual data and using LLM-based as well. The results show that the composite datasets can increase the number of data at a low cost and improve the performance of downstream tasks under all budgets.

Addressing the Imbalanced Class Issue. Figure 3 shows that there is a difference in the number of annotations for each entity type. This imbalance issue is even greater with annotation by LLM. Also, the performance difference of LLM-based annotation per entity type is large compared to manual annotation. In terms of the performance difference between labels, the variance of F1-scores when using LLM-based annotation is larger than that of manual annotation. Label mixing reduced the performance difference between entity types. Therefore, we believe that soft labels, according to possible entities, will prevent model training in a biased manner, unlike one-hot labels.

Table 3 presents a detailed quantification of these

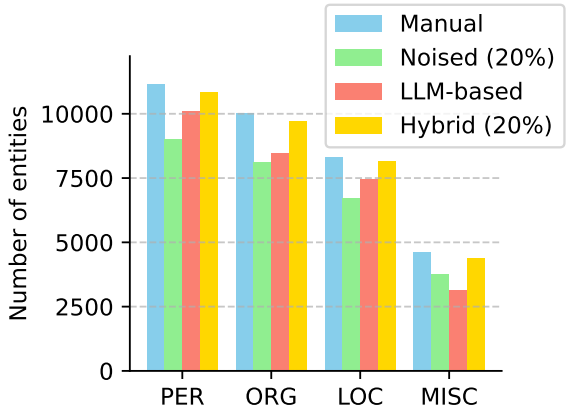

Figure 3: Number of annotations for each entity type in the manual, noised, and LLM-based datasets.

	CoNLL03 ORG	CoNLL03 LOC	CoNLL03 MISC	WikiGold ORG	WikiGold LOC	WikiGold MISC
PER	72	37	15	13	5	1
ORG	-	180	59	-	44	41
LOC	-	-	158	-	-	5

Table 3: The number of entities overlapping among classes in the CoNLL03 and WikiGold dataset.

overlapping entities between different entity types when annotated by LLM. Moreover, the fact that the number of entities overlapping between classes such as LOC and MISC with ORG is comparatively higher than other combinations, while there is less overlap between PER and MISC, indicates a potential challenge in the discriminatory capability of LLM-based annotations for the former combinations. These observations support the effectiveness of label mixing as a technique to buffer these shortcomings, suggesting that it serves as a valuable strategy for improving the robustness and performance of NER models.

5 Discussion and Conclusion

Our study extends the use of LLMs toward the annotation process. We addressed NER annotation challenges of missing labels and proposed novel solutions. Our integration of LLM-based annotation with manual annotation significantly enhances performance of NER task, especially when using noised datasets or in limited budgets. We have shown that our proposed label mixing address the performance degradation due to biases and imbalances. Our approach capitalizes on the capabilities of LLMs, which are expected to continue advancing in the future (Zhao et al., 2023; Min et al., 2023).

6 Ethical Considerations

In our research, we focus not on generating text, but on the automation of annotations. This approach places concerns about creating potentially harmful sentences outside our scope of study. However, while our primary objective is to annotate entities with labels, it is essential to consider the implications of certain annotations, such as the toxicity of entities, especially when these annotations pertain to specific demographic groups.

This risk stems from biases inherent in the LLMs used for annotation, necessitating continuous monitoring.

7 Limitations

Finally, we would like to mention the limitations of our work. As the first limitation, we have only validated our proposed methods with relatively small datasets, CoNLL03 and Wikigold, thus requiring validation with larger NER datasets, such as OntoNotes 5.0 (Weischedel et al., 2013).

Another limitation is that when using the ChatGPT (gpt-35-turbo-0613) API, some regions and organizations are not generated due to their alignment. It is conceivable that this kind of alignment-based distribution bias could affect NER task performance.

The final limitation involves the LLM-based annotation method for NER tasks, which requires computational costs proportional to the number of entity types. Thus, algorithms independent of entity type count are needed.

References

Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. 2009. Named entity recognition in Wikipedia. In *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources (People's Web)*, pages 10–18, Suntec, Singapore. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. 2023. Is GPT-3 a good data annotator? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11173–11195, Toronto, Canada. Association for Computational Linguistics.

Karën Fort, Gilles Adda, and Kevin Bretonnel Cohen. 2011. Amazon mechanical turk: Gold mine or coal mine? *Computational Linguistics*, pages 413–420.

Sreyan Ghosh, Utkarsh Tyagi, Manan Suri, Sonal Kumar, S Ramaneswaran, and Dinesh Manocha. 2023. Aclm: A selective-denoising based generative data augmentation approach for low-resource complex ner. *arXiv preprint arXiv:2306.00928*.

Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023a. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *ArXiv*, abs/2305.14450.

Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023b. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *Preprint*, arXiv:2305.14450.

Arzoo Katiyar and Claire Cardie. 2018. Nested named entity recognition revisited. In *North American Chapter of the Association for Computational Linguistics*.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56(2):1–40.

Shubhanshu Mishra, Sijun He, and Luca Belli. 2020. Assessing demographic bias in named entity recognition. *arXiv preprint arXiv:2008.03415*.

Ruslan Mitkov, Constantin Orasan, and Richard J. Evans. 2000. The importance of annotated corpora for nlp: the cases of anaphora resolution and clause splitting.

Behrang Mohit. 2014. Named entity recognition. In *Natural language processing of semitic languages*, pages 221–245. Springer.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023a. Diffusionner: Boundary diffusion for named entity recognition. *arXiv preprint arXiv:2305.13298*.

Yongliang Shen, Zeqi Tan, Shuhui Wu, Wenqi Zhang, Rongsheng Zhang, Yadong Xi, Weiming Lu, and Yueting Zhuang. 2023b. Promptner: Prompt locating and typing for named entity recognition. *arXiv preprint arXiv:2305.17104*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In

Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pages 142–147.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models. *ArXiv*, abs/2304.10428.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2013. OntoNotes Release 5.0.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*.

Shuai Zhang, Yongliang Shen, Zeqi Tan, Yiquan Wu, and Weiming Lu. 2022. De-bias for generative extraction in unified ner task. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 808–818.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Appendix

A Details of Experiment Setup

A.1 Hardware Configurations:

In our experiments, we used virtual machines on Google Cloud. The detailed machine specifications are as follows:

- Machine type: `n1 standard 4`
- Memory: 15 GB
- virtual CPU: `Intel Broadwell` x 4
- GPU: `NVIDIA T4` x 1

A.2 BERT Configurations:

The configurations of BERT used for the performance evaluation are as follows:

- Pre-trained model: `bert-model-unbased` [2]
- Learning rate: $1e-05$
- Upper limit for norm of gradients: 10
- Batch size: 64 for training, 32 for inference
- Max length of token: 128

Each training was terminated when the F1 score on the validation set remained unchanged for three consecutive epochs.

In NER, labels must be assigned at the word level, but BERT's tokenizer often splits a single word into multiple tokens. Therefore, in this study, when a word is split into multiple tokens, when a word is split into multiple tokens, the model adopts the label for the first token as the label for the entire word.

A.3 Code Implementation

The experimental code for this study was implemented using Copilot.[3]

B Datasets

B.1 CoNLL03 (Tjong Kim Sang and De Meulder, 2003):

The CoNLL-2003 dataset is designed to recognize four types of named entities, including person names, place names, organization names, and other miscellaneous entities. The data file consists of four columns: word, part-of-speech tag, syntactic chunk tag, and named entity tag, with each item separated by a space. The named entity and syntactic chunk tags use the I-TYPE format [4], and the B-TYPE [5] tag is introduced to indicate the start of a new phrase.

B.2 WikiGold (Balasuriya et al., 2009):

The Wikigold dataset (Balasuriya et al., 2009) is a manually annotated corpus for named entity recognition consisting of a small sample of Wikipedia articles. Words are labeled for each of the four named entity classes (LOC, MISC, ORG, and PER) of CONLL-03 described above.

C Statistics of Overlapped Entities

LLM-based annotations can overlap different classes on the same entity. We investigate the statistics of overlapped entities.

The fact that the entities overlapping between classes constitute only 521 cases (2.65%) of the CoNLL03 dataset and 109 cases (6.80%) of the WikiGold dataset, and yet the application of label mixing to these specific data points results in a performance improvement across the dataset, is a suggestive observation. This observation implies that the strategic application of label mixing to even a small subset of data can improve the performance balance between entity types.

[2] https://huggingface.co/bert-base-uncased

[3] `https://github.com/features/copilot`

[4] This tag means "Inside" and is assigned to words that are within an entity of a specific type (TYPE). That is, this tag is used when a word is part of a specific entity and indicates the type of that entity. For example, "I-PER" indicates that the word is part of a person's name (Person).

[5] The B in B-TYPE means "Beginning" and is the tag given to the first word of the second entity when directly followed by an entity of the same type. This tag is used to indicate that a new entity is beginning and to distinguish it from the previous entity. For example, "B-PER" means that the new person's name begins with this word.