muNERa at WojoodNER 2024: Multi-tasking NER Approach

Nouf M. Alotaibi [1] **Haneen Alhomoud** [1] **Hanan Murayshid** [2]

Waad Alshammari [3] **Nouf Alshalawi** [2] **Sakhar Alkhereyf** [1,2]

[1] Saudi Data and AI Authority (SDAIA), Saudi Arabia
[2] King Abdulaziz City for Science and Technology (KACST), Saudi Arabia
[3] King Salman Global Academy for Arabic Language (KSAA), Saudi Arabia
nomotaibi,hmomoud@sdaia.gov.sa , salkhereyf@kacst.gov.sa

Abstract

This paper presents our system "muNERa", submitted to the WojoodNER 2024 shared task at the second ArabicNLP conference. We participated in two subtasks, the flat and nested fine-grained NER sub-tasks (1 and 2). muNERa achieved first place in the nested NER sub-task and second place in the flat NER sub-task. The system is based on the TANL framework (Paolini et al., 2021), by using a sequence-to-sequence structured language translation approach to model both tasks. We utilize the pre-trained AraT5$_{v2}$-base model as the base model for the TANL framework. The best-performing muNERa model achieves 91.07% and 90.26% for the F-1 scores on the test sets for the nested and flat subtasks, respectively.

1 Introduction

The abundance of written texts in the digital age has increased the importance of identifying key pieces of information, known as named entities, within the text. This process, called Named Entity Recognition (NER), involves identifying and classifying entities into pre-defined categories, such as names of persons, organizations, locations, dates, and more, depending on the application. The ability to accurately identify these entities is crucial for numerous real-time applications, including biomedical and clinical studies (Liu et al., 2022), business (Shah et al., 2023), and law (Kalamkar et al., 2022). Moreover, NER complements other NLP tasks, including Question Answering (Guven and Unalir, 2021), Machine Translation (Mota et al., 2022), and Coreference Resolution (Wang and El-Gohary, 2023).

This paper presents our system, muNERa, which participated in the WojoodNER 2024 shared task at the second ArabicNLP conference (Jarrar et al., 2024). The WojoodNER shared task, now in its second year, includes two subtasks: flat and nested fine-grained NER. Also, there is a third "open track" subtask in which we did not participate. WojoodNER aims to advance the state-of-the-art in Arabic NER by providing a large-scale, fine-grained, and annotated dataset and a competitive platform for researchers to evaluate their models. The main difference between WojoodNER 2024 and 2023 is the introduction of *Wojood$_{Fine}$* with finer-grained sub-tags and sub-sub-tags (Liqreina et al., 2023). Also, there has been a major revision of the tagging schemes.

Typically, NER is modeled as a sequence classification problem such that each token is assigned a tag using a sequence tagging scheme such as IOB2 (Tjong Kim Sang and Veenstra, 1999). In muNERa, we model both the nested and flat NER tasks as a sequence-to-sequence translation utilizing the Translation between Augmented Natural Languages (TANL) framework (Paolini et al., 2021). Notably, all participating teams in the Wojood 2023 shared task (Jarrar et al., 2023) did not utilize sequence-to-sequence (encoder-decoder) architecture, their work employed BERT pre-trained language models (encoder-only) in different ways (Ehsan et al., 2023; Laouirine et al., 2023; El Mahdaouy et al., 2023; Li et al., 2023; Elkordi et al., 2023).

2 Work on WojoodNER 2023

WojoodNER-2023, the first Arabic NER shared task (Jarrar et al., 2023), includes FlatNER and NestedNER subtasks with AraBERTv2 as the baseline (Antoun et al., 2020). Notably, most participating teams utilized BERT pre-trained language models, with the performance of the encoder-decoder T5 model remaining unexplored (Raffel et al., 2023). Seven studies employed AraBERT (Antoun et al., 2020), and it is remarkable that AraBERT consistently outperformed other transformers: LIPN (El Elkhbir et al., 2023) achieved first place in FlatNER, ELYADATA (Laouirine

Proceedings of The Second Arabic Natural Language Processing Conference, pages 858–866

et al., 2023) excelled in NestedNER. El-Kawaref (Elkaref and Elkaref, 2023) utilized StagedNER, while UM6P & UL (El Mahdaouy et al., 2023) adopted multi-task learning. Additionally, Lotus (Li et al., 2023) explored multi-task learning with XLM-R, AraBERT, and MARBERT, favoring XLM-R. AlexU-AIC (Elkordi et al., 2023) employed a machine reading comprehension-based approach, leveraging various PLMs, while Alex-U 2023 NLP (Hussein et al., 2023) introduced AraBINDER, utilizing BERT models.

In our work, we explore a new method utilizing the T5 encoder-decoder architecture instead of the BERT encoder architecture.

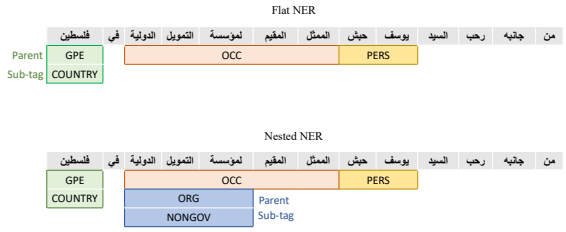

Figure 1: An example of the hierarchy tagging in flat and nested NER on Wojood Fine-Grained Corpus.

3 Data

The Wojood Corpus is a dataset for Arabic NER dataset containing approximately 550,000 tokens of Modern Standard Arabic and dialects. This dataset is designed specifically to identify nested entities, where entities are embedded within other entities. The Wojood Corpus consists of two versions for Arabic nested NER: the original and the $\text{Wojood}_{\text{Fine}}$ Corpus (Jarrar et al., 2022; Liqreina et al., 2023).

3.1 $\text{Wojood}_{\text{Fine}}$ Corpus

$\text{Wojood}_{\text{Fine}}$ (Liqreina et al., 2023) is a comprehensive Arabic corpus for nested NER focusing on fine-grained entity types across various domains. This refined extension of the original Wojood Corpus (Jarrar et al., 2022) contains tokens annotated across various entity types, including geopolitical entities (GPE), locations (LOC), organizations (ORG), and facilities (FAC). Each entity type contains sub-types from 31 sub-types following the LDC's ACE guidelines (Walker et al., 2005). This dataset features 47.6K fine-grained and nested annotations, allowing for more detailed entity recognition than provided by standard NER tasks. Table 5 in Appendix A details the frequencies of the 53 NER tags present in the training and development data splits.

3.2 Data Preprocessing

As detailed in section 4, we adapted the $\text{Wojood}_{\text{Fine}}$ Corpus to meet the input requirements of the TANL framework (Paolini et al., 2021) to use it for modeling effectively. The preprocessing steps involve extracting hierarchical tags (parent, sub-tag, sub-subtag) and their spans using the BIO scheme. Then, each token and its corresponding labels are reformatted to align with the TANL framework's specifications. For example, the sentence "رسالة إلى المندوب السامي في فلسطين." will be, before preprocessing (using the BIO scheme),: [(رسالة, O), (إلى, O), (المندوب, B-OCC), (السامي, I-OCC), (في, I-OCC), (فلسطين, B-GPE, I-OCC), (., O)] ; and after preprocessing: tokens: [رسالة، إلى، المندوب، السامي، في، فلسطين، .], entities: [type: GPE, start: 5, end: 6, type: OCC, start: 2, end: 6].

3.3 Challenges

This subsection discusses some challenges we faced while working on the $\text{Wojood}_{\text{Fine}}$ Corpus.

Complexity of the Tagging System: The tagging architecture employed within the $\text{Wojood}_{\text{Fine}}$ Corpus introduces a three-level nested tagging scheme that significantly complicates the annotation and recognition processes. In this scheme, a single token can be associated with many different parent tags (in the case of nested tagging). Each parent tag may have a sub-tag as shown in Figure 1, and each sub-tag can include up to two sub-sub tags. This results in a multi-layered tagging structure where a single token can be annotated with many tags across the three levels. In fact, in the $\text{Wojood}_{\text{Fine}}$ Corpus, some tokens have up to five different parent tags and eight subtags.

This complexity increases the difficulty of accurately tagging the data and poses substantial challenges for the NER models in learning and predicting such a diverse range of tag combinations effectively.

Inconsistency in Tag Distribution: Some tags, such as the sub-tag "ENT", were present in the development set but not in the training set. This inconsistency can create significant challenges during model training, as the model may not learn

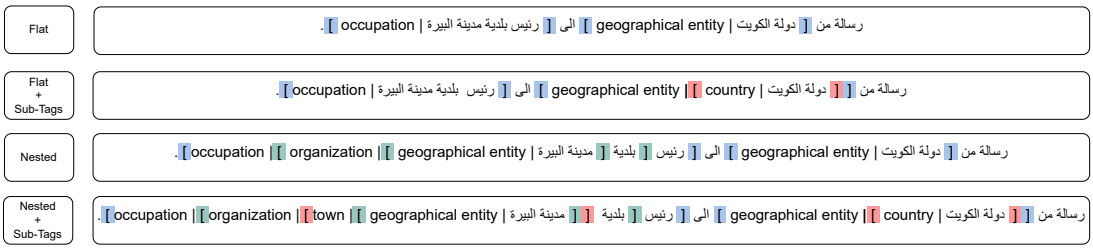

Figure 2: Illustration of TANL Format for flat and nested Wojood dataset. The blue highlighted brackets represent the span of parent tags, while green highlights are for nested parent tags, and red highlights are for sub-tags.

how to recognize entities that only appear in the development or test set but not in the training set.

Sparse Examples for Certain Tags: The dataset shows that several tags are rarely represented. For example, the parent tag "UNIT" only appears six times in the training set. Similarly, the sub-tag "LAND" appears only once each. Another example is the tag "Path", which differs from "PATH" due to case sensitivity and has only one example in the training set. This issue also extends to sub-sub tags like "ENT" and "PLANT", which have only one and three examples, respectively. Having a few number of examples for some tags makes it difficult for models to recognize them accurately.

4 Methodology

We utilize the TANL framework (Paolini et al., 2021) to train our models for both flat and nested NER. The TANL framework is designed to address structured prediction tasks in language processing. These tasks include joint entity and relation extraction, relation classification, semantic role labeling, event extraction, coreference resolution, dialogue state tracking, and nested named entity recognition (Paolini et al., 2021).

One key advantage of the TANL framework is its ability to incorporate the semantic understanding of labels during model training. Unlike traditional task-specific classifiers, which typically train without explicit knowledge of label semantics, TANL leverages pre-trained models' understanding of entity semantics (e.g., Person, Location). In the TANL framework, depending on the task, both the input and output are structured in augmented natural languages. In our case, we created the NER structure task in TANL similar to the Adverse Drug Effect (ADE) dataset (Gurulingappa et al., 2012) structure task, as both datasets feature nested NER entities. The task aims to extract entities and their spans from a given sentence. These entities are enclosed within the special tokens **[ ]**, and each entity is followed by its type and separated by **|**, e.g., **[token | entity type]**. Entity types are represented in natural language words—such as person or location—rather than abbreviations like PER or LOC to leverage the model's semantic knowledge of these words. For WojoodNER labels, we manually created a dictionary mapping the entity tags (e.g., PER) to their natural language words (e.g., person) based on the description of the tags in the $\text{Wojood}_{\text{Fine}}$ paper (Liqreina et al., 2023). Some entity tags are already in a natural language format, such as "PRODUCT" and "SPORT", while others require extracting the appropriate natural language terms from their descriptions as shown in Table 1.

Tag	Natural Language
LOC	location
GOV	government
PRODUCT	product
SPO	sports organizations
SPORT	sport

Table 1: Examples of WojoodNER tags and their corresponding natural language words.

For Nested entities, the task allows the representing of entity hierarchies, for example **[token [token | entity type$_1$] | entity type$_2$]**. The sub-tag and sub-sub-tags were treated the same as nested parent tags. In other words, a span with multiple tags (parents and subtags) will be treated as the whole span is nested within itself using the other tags. Figure 2 illustrates how the Wojood dataset is formatted as input and output using the TANL framework.

- For flat: entities **دولةالكويت** and **رئيس بلدية مدينة البيرة** tagged as "geographical entity" and "occupation", respectively.
- For flat with sub-tags: entity **دولةالكويت** has

a sub-tag "country".

- For Nested: entity رئيس بلدية مدينة البيرة has multiple parent tags, including بلدية مدينة البيرة and مدينة البيرة tagged as "organization" and "geographical entity", respectively.
- For Nested with sub-tags: entities مدينة البيرة and دولةالكويت have sub-tags "town" and "country", respectively.

After generating the output, misspellings of the tokens in the original text or minor format errors can occur due to the nature of generative models. To address this issue, TANL's decoding process first cleans the output by removing special tokens and discarding invalid formats, ensuring only valid entity types are considered. To further enhance robustness, TANL utilizes the dynamic programming (DP) based Needleman-Wunsch alignment algorithm (Needleman and Wunsch, 1970), which aligns input and cleaned output tokens at the token level. For instance, it can correctly align a misspelled generated token (e.g., مدينة البيرا) with its correct form in the input (e.g., مدينة البيرة), ensuring accurate entity identification. We noticed that most of the invalid outputs are:

- Invalid format, such as missing open bracket "[" or the closed bracket "]" tokens. For example, the generated output "عمران] الادلة على [[مدينة قلقيلية | town] | geographical entity] | facility] | building or grounds] ..."' has a missing open bracket "[" token. In this case, the last entity "building or grounds" is discarded before the alignment process.
- Missing the end-of-sentence period ".", and repeating some tokens, such as "رسالة رئيس بلدية بيت لحم للحاكم العسكري في منطقة بيت لحم للحاكم العسكري في منطقة بيت لحم" instead of the original text "رسالة رئيس بلدية بيت لحم للحاكم العسكري في منطقة بيت لحم".
- Additionally, some invalid inputs include generating other variations of Arabic dialects. For example, the original text "فيروز : شوفي شوفي بكلم بحاله ." was reconstructed as "فيروز : شوفي شوفي بكلم لحاله.".

In our case, although there were a few examples where the entities in invalid format text, such as missing the open "[" or closed "]" bracket, were discarded, all of the invalid reconstructed examples in the development and test sets were successfully aligned.

We use the AraT5$_{v2}$ model (Elmadany et al., 2023) as the base for the TANL framework. AraT5$_{v2}$ is a pre-trained Arabic model based on the T5 (Text-to-Text Transfer Transformer model) encoder-decoder transformer architecture (Raffel et al., 2020). We utilize TANL based on AraT5$_{v2}$ as a multi-tasking span detector and entity recognition such that it outputs the structured predictions of spans and their entity types. We use two distinct TANL models: one for flat NER and one for nested NER. We also use fastText (FT) classifier (Joulin et al., 2016) as a secondary tagger such that we first use TANL to detect spans and assign level-1 tags (parent tags), then we tag the detected span with level 2 and 3 using the FT classifier.

Evaluation metrics: We follow the WojoodNER shared task evaluation criteria and report the micro-average scores for Precision (P), Recall (R), and F-1 (F). Additionally, we report the F-1 score for span detection, where every span in the evaluation set is considered positive regardless of its type.

5 Experiments and Results

We use the TANL model (Paolini et al., 2021), based on AraT5$_{v2}$ (Elmadany et al., 2023) as the primary classifier, and fastText (FT) (Joulin et al., 2016) trained on spans from the training set as a secondary classifier. For FT, we use the pre-Arabic trained vectors *fasttext-ar-vectors* (Grave et al., 2018), which is available on HuggingFace [1]. For TANL, we set the values for hyperparameters as follows:

Hyperparameter	Value
Learning rate (LR)	$5e-4$
Beam Size	8
Number of epochs	5
Train Batch size	4
Max sequence length	768

To train the TANL model, we use Microsoft Azure ML service with the NC A100 v4 series, powered by Nvidia A100 GPUs with 80 GB memory. For inference, we use Google Colab Pro+ with one Nvidia A100 GPU.

[1] https://huggingface.co/facebook/fasttext-ar-vectors

		Entity Recognition			Span Det.
Model	**Task**	**P**	**R**	**F-1**	**F-1**
TANLp	N	**92.41**	58.86	71.91	91.90
TANLp + FT	N	90.74	90.01	90.37	
TANLs	N	92.44	89.91	**91.16**	**92.05**
TANLs + FT	N	90.08	**90.15**	90.11	
TANLp	F	89.44	57.75	70.19	**89.32**
TANLp + FT	F	88.32	84.1	86.16	
TANLs	F	**91.16**	89.96	**90.55**	87.54
TANLs + FT	F	84.84	**90.68**	87.66	

Table 2: Results of our models on the validation set. N is the nested subtask, and F is the flat subtask. TANLp is TANL with parent tags only (level 1); and TANLs with subtags (all tags); FT is fastText. P: precision; R: recall.

Table 2 shows the results on the validation set for different models. TANLp is a TANL model trained only on level-1 tags (parent tags), and TANLs is trained on all levels (i.e., parent tags, sub-tags, and sub-sub-tags). (+ FT) indicates additional tagging of the detected spans using the FT model with the remaining levels (i.e., sub-tags and sub-sub-tags).

The results indicate that the best model on both tasks (i.e., nested and flat) is TANLs, which tags all levels without additional tagging using the secondary classifier FT. We observe that all TANL-based models perform well in span detection and recognition. Additionally, when using fastText as a secondary tagger with TANLp, the results indicate that the performance is close to TANLs for nested (less than 1% difference in F-1), but not the case with the flat subtask. This may be due to over-predicting entity types by fastText, as reflected in the precision and recall scores.

For TANL with the simpler tagging scheme (i.e., TANLp), the model performs better than the model with subtags in span detection for the flat subtask. However, for the nested subtask, the model trained on subtags performs better than the model trained on only parent tags, TANLp.

5.1 System Submissions

Based on the validation set results shown in Table 2, we select TANLs as the best model for both tasks: nested and flat. We report the performance on the blind test set in Table 3, showing the rank of our system among other submissions to the two subtasks.

Rank	Model	Task	P	R	F-1
1	mucAI	F	**91**	90	**90**
2	**muNERa**	F	**91**	89	**90**
2	Addax	F	89	**91**	**90**
	baseline	F	89	90	89
3	DRU	F	86	88	87
4	Bangor	F	88	85	86
	baseline	N	92	93	92
1	**muNERa**	N	**92**	**90**	**91**
2	DRU	N	90	90	90

Table 3: Results on the blind test set for all submissions to the two subtasks (F: flat; N: Nested). P: precision, R: recall. All scores are micro-averaged for all classes.

6 Conclusions

This paper presents the results of "muNERa" on the WojoodNER 2024 shared task. The muNERa system demonstrated outstanding performance in the WojoodNER 2024 shared task, securing first place in the Nested NER sub-task and second place in the Flat NER sub-task. The system's success is primarily attributed to the innovative application of the TANL framework (Paolini et al., 2021), combined with the AraT5v2-base model (Elmadany et al., 2023). These methodologies allowed muNERa to achieve remarkable F-1 scores of 91.07% for Nested NER and 90.26% for Flat NER.

Our system shows that the encoder-decoder architecture tackles the complexity of the $\text{Wojood}_{\text{Fine}}$ Corpus annotation scheme, effectively modeling the multi-layered tagging scheme and the challenge of the nested entities structure. Despite the challenges posed by inconsistencies and sparse examples in the dataset, muNERa consistently delivered high-precision and recall metrics.

For future work, there are several directions. One is integrating the hierarchical structure of the 3-level tagging scheme (i.e., tags, sub-tags, and sub-sub-tags) into the model, potentially improving performance. An example is experimenting with different structures for the TANL format that incorporate the hierarchical tagging scheme. Additionally, employing more advanced secondary taggers, like BERT, could improve the performance further than our secondary tagger, fastText. Finally, addressing the computational limitations and experimenting with a broader range of hyperparameters will also be crucial for future advancements.

7 Limitations

There are some limitations regarding the design of muNERa for the WojoodNER 2024 shared task.

- The system does not consider the hierarchical structure of the 3-level tagging scheme. This can be a future direction, such that the model considers sub-tags and sub-sub-tags to be associated with certain main tags.
- We use two distinct TANL models, one for each sub-task (i.e., nested and flat). Training a single model on both tasks, allowing the task to be prompted in the input, is a future research direction.
- We have shown in subsection 3.3 some challenges in the $\text{Wojood}_{\text{Fine}}$ dataset. One is the class imbalance challenge. Also, we have shown that some tags were present in the development and test set but not in the training set. Due to computational and time constraints, we haven't experimented with data resampling techniques.
- We have used English natural language words for entities as described in section 4. We have not tried to use Arabic words. We believe this would be a good direction to enhance the model, especially since our base model, AraT5_{v2}, was pre-trained mainly on Arabic datasets.
- We have used fastText as a secondary tagger with TANL as the primary one. We did not use a more advanced secondary tagger, such as BERT. This can be a future direction where more advanced models can be used as secondary taggers.
- Due to computational limits, we have not experimented with a wider range of hyperparameters, such as the number of epochs or beam size for TANL.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and feedback. We also thank SDAIA for providing access to Microsoft Azure Service to train the models and KSAA for providing Google Colab Pro+ accounts to evaluate and test the models used in this paper. Finally, we would like to thank the WojoodNER 2024 shared task organizers for organizing the shared task and for their prompt and helpful responses to our inquiries.

References

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Toqeer Ehsan, Amjad Ali, and Ala Al-Fuqaha. 2023. AlphaBrains at WojoodNER shared task: Arabic named entity recognition by using character-based context-sensitive word representations. In *Proceedings of ArabicNLP 2023*, pages 783–788, Singapore (Hybrid). Association for Computational Linguistics.

Niama El Elkhbir, Urchade Zaratiana, Nadi Tomeh, and Thierry Charnois. 2023. LIPN at WojoodNER shared task: A span-based approach for flat and nested Arabic named entity recognition. In *Proceedings of ArabicNLP 2023*, pages 789–796, Singapore (Hybrid). Association for Computational Linguistics.

Abdelkader El Mahdaouy, Salima Lamsiyah, Hamza Alami, Christoph Schommer, and Ismail Berrada. 2023. UM6P & UL at WojoodNER shared task: Improving multi-task learning for flat and nested Arabic named entity recognition. In *Proceedings of ArabicNLP 2023*, pages 777–782, Singapore (Hybrid). Association for Computational Linguistics.

Nehal Elkaref and Mohab Elkaref. 2023. El-kawaref at WojoodNER shared task: StagedNER for Arabic named entity recognition. In *Proceedings of ArabicNLP 2023*, pages 803–808, Singapore (Hybrid). Association for Computational Linguistics.

Shereen Elkordi, Noha Adly, and Marwan Torki. 2023. AlexU-AIC at WojoodNER shared task: Sequence labeling vs MRC and SWA for Arabic named entity recognition. In *Proceedings of ArabicNLP 2023*, pages 771–776, Singapore (Hybrid). Association for Computational Linguistics.

AbdelRahim Elmadany, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Octopus: A multitask model and toolkit for Arabic natural language generation. In *Proceedings of ArabicNLP 2023*, pages 232–243, Singapore (Hybrid). Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5):885–892. Text Mining and Natural Language Processing in Pharmacogenomics.

Zekeriya Anil Guven and Murat Osman Unalir. 2021. Improving the bert model with proposed named entity recognition method for question answering. *2021 6th International Conference on Computer Science and Engineering (UBMK)*, pages 204–208.

Mariam Hussein, Sarah Khaled, Marwan Torki, and Nagwa El-Makky. 2023. Alex-U 2023 NLP at WojoodNER shared task: AraBINDER (bi-encoder for Arabic named entity recognition). In *Proceedings of ArabicNLP 2023*, pages 797–802, Singapore (Hybrid). Association for Computational Linguistics.

Mustafa Jarrar, Muhammad Abdul-Mageed, Mohammed Khalilia, Bashar Talafha, AbdelRahim Elmadany, Nagham Hamad, and Alaa' Omar. 2023. WojoodNER 2023: The first Arabic named entity recognition shared task. In *Proceedings of ArabicNLP 2023*, pages 748–758, Singapore (Hybrid). Association for Computational Linguistics.

Mustafa Jarrar, Nagham Hamad, Mohammed Khalilia, Bashar Talafha, and Muhammad Elmadany, Abdel-Rahim Abdul-Mageed. 2024. WojoodNER 2024: The Second Arabic Named Entity Recognition Shared Task. In *Proceedings of the 2nd Arabic Natural Language Processing Conference (Arabic-NLP), Part of the ACL 2024.* Association for Computational Linguistics.

Mustafa Jarrar, Mohammed Khalilia, and Sana Ghanem. 2022. Wojood: Nested Arabic named entity corpus and recognition using BERT. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3626–3636, Marseille, France. European Language Resources Association.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Prathamesh Kalamkar, Astha Agarwal, Aman Tiwari, Smita Gupta, Sau Meng Karn, and Vivek Raghavan. 2022. Named entity recognition in indian court judgments. *ArXiv*, abs/2211.03442.

Imen Laouirine, Haroun Elleuch, and Fethi Bougares. 2023. ELYADATA at WojoodNER shared task: Data and model-centric approaches for Arabic flat and nested NER. In *Proceedings of ArabicNLP 2023*, pages 759–764, Singapore (Hybrid). Association for Computational Linguistics.

Jiyong Li, Dilshod Azizov, Hilal AlQuabeh, and Shangsong Liang. 2023. Lotus at WojoodNER shared task: Multilingual transformers: Unveiling flat and nested entity recognition. In *Proceedings of ArabicNLP 2023*, pages 765–770, Singapore (Hybrid). Association for Computational Linguistics.

Haneen Liqreina, Mustafa Jarrar, Mohammed Khalilia, Ahmed El-Shangiti, and Muhammad Abdul Mageed. 2023. Arabic fine-grained entity recognition. In *Proceedings of ArabicNLP 2023*, pages 310–323.

Ning Liu, Qian Hu, Hu Shan Xu, Xing Xu, and Mengxin Chen. 2022. Med-bert: A pretraining framework for medical records named entity recognition. *IEEE Transactions on Industrial Informatics*, 18:5600–5608.

Pedro Mota, Vera Cabarrão, and Eduardo Farah. 2022. Fast-paced improvements to named entity handling for neural machine translation. In *European Association for Machine Translation Conferences/Workshops*.

Saul B Needleman and Christian D Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3):443–453.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. *Preprint*, arXiv:2101.05779.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Agam Shah, Ruchit Vithani, Abhinav Gullapalli, and Sudheer Chava. 2023. Finer: Financial named entity recognition dataset and weak-supervision model. *ArXiv*, abs/2302.11157.

Erik F. Tjong Kim Sang and Jorn Veenstra. 1999. Representing text chunks. In *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 173–179, Bergen, Norway. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2005. Ace 2005 multilingual training corpus-linguistic data consortium. *URL: https://catalog. ldc. upenn. edu/LDC2006T06*.

Xiyu Wang and Nora M. El-Gohary. 2023. Deep learning-based named entity recognition and resolution of referential ambiguities for enhanced information extraction from construction safety regulations. *J. Comput. Civ. Eng.*, 37.

A Detailed results

Table 4 shows the details of results for the best model for nested NER in section 5 (TANLs). Table 5 shows the distribution of entity classes in the train and development sets.

Class	P	R	F-1
BOUNDARY	50.00	50.00	50.00
BUILDING -OR-GROUNDS	81.19	80.39	80.79
CAMP	100.00	97.18	98.57
CARDINAL	79.75	76.47	78.08
CLUSTER	85.71	66.67	75.00
COM	77.78	70.00	73.68
CONTINENT	92.00	100.00	95.83
COUNTRY	98.67	97.84	98.25
CURR	94.74	75.00	83.72
DATE	94.67	93.49	94.08
EDU	90.82	80.18	85.17
ENT	0.00	0.00	0.00
EVENT	83.61	69.86	76.12
FAC	79.46	80.18	79.82
GOV	84.63	78.85	81.64
GPE	98.43	96.82	97.62
GPE_ORG	85.03	85.03	85.03
LAND-REGION -NATURAL	77.42	92.31	84.21
LANGUAGE	76.47	81.25	78.79
LAW	80.39	87.23	83.67
LOC	83.33	80.15	81.71
MED	99.52	98.81	99.16
MONEY	70.59	54.55	61.54
NEIGHBORHOOD	60.00	60.00	60.00
NONGOV	91.36	90.88	91.12
NORP	73.08	70.96	72.01
OCC	86.80	83.30	85.01
ORDINAL	95.65	93.01	94.32
ORG	94.43	92.48	93.44
ORG_FAC	50.00	42.86	46.15
PATH	50.00	33.33	40.00
PERCENT	76.92	83.33	80.00
PERS	92.61	92.34	92.48
PRODUCT	66.67	25.00	36.36
QUANTITY	50.00	66.67	57.14
REGION-GENERAL	84.21	84.21	84.21
REGION -INTERNATIONAL	76.92	76.92	76.92
REL	64.29	90.00	75.00
SCI	81.82	69.23	75.00
SPO	100.00	50.00	66.67
SPORT	100.00	100.00	100.00
STATE-OR -PROVINCE	92.98	88.83	90.86
SUBAREA -FACILITY	91.67	68.75	78.57
TIME	68.75	66.67	67.69
TOWN	97.08	95.40	96.23
UNIT	75.00	75.00	75.00
WATER-BODY	100.00	61.54	76.19
WEBSITE	66.67	45.00	53.73
micro avg	**92.44**	**89.91**	**91.16**
macro avg	**80.23**	**74.96**	**76.80**

Table 4: The Precision (P), Recall (R), and F-1 scores per each class in the development set for the TANLs model on the nested subtask.

Tag	Flat			Nested		
	Train	**Dev**	**Total**	**Train**	**Dev**	**Total**
AIRPORT	12	0	**12**	13	0	**13**
BOUNDARY	46	12	**58**	46	12	**58**
BUILDING-OR-GROUNDS	1566	226	**1792**	1654	244	**1898**
CAMP	1379	164	**1543**	1402	167	**1569**
CARDINAL	1641	200	**1841**	1669	200	**1869**
CELESTIAL	2	0	**2**	2	0	**2**
CLUSTER	335	41	**376**	476	59	**535**
COM	1245	107	**1352**	1248	108	**1356**
CONTINENT	68	10	**78**	136	23	**159**
COUNTRY	3445	532	**3977**	6320	936	**7256**
CURR	18	6	**24**	214	32	**246**
DATE	50055	7380	**57435**	50930	7512	**58442**
EDU	1178	130	**1308**	1944	249	**2193**
ENT	1	2	**3**	1	2	**3**
EVENT	5957	871	**6828**	6125	901	**7026**
FAC	1534	221	**1755**	1806	263	**2069**
GOV	12410	1859	**14269**	12543	1875	**14418**
GPE	13810	1917	**15727**	23812	3415	**27227**
GPE_ORG	1274	209	**1483**	1324	217	**1541**
LAND	1	0	**1**	1	0	**1**
LAND-REGION-NATURAL	305	41	**346**	328	45	**373**
LANGUAGE	143	18	**161**	144	18	**162**
LAW	1365	177	**1542**	1365	177	**1542**
LOC	1646	241	**1887**	2031	290	**2321**
MED	6260	914	**7174**	6260	914	**7174**
MONEY	483	58	**541**	483	58	**541**
NEIGHBORHOOD	214	15	**229**	228	15	**243**
NONGOV	11654	1585	**13239**	11753	1599	**13352**
NORP	6508	922	**7430**	7095	995	**8090**
OCC	11716	1684	**13400**	11993	1734	**13727**
ORDINAL	3085	511	**3596**	3791	611	**4402**
ORG	26927	3760	**30687**	33043	4622	**37665**
ORG_FAC	286	26	**312**	286	26	**312**
PATH	155	10	**165**	155	10	**165**
PERCENT	233	28	**261**	233	28	**261**
PERS	9233	1206	**10439**	9983	1291	**11274**
PLANT	3	0	**3**	3	0	**3**
PRODUCT	157	23	**180**	160	23	**183**
Path	1	0	**1**	1	0	**1**
QUANTITY	96	6	**102**	99	6	**105**
REGION-GENERAL	703	97	**800**	709	97	**806**
REGION-INTERNATIONAL	143	25	**168**	149	25	**174**
REL	202	33	**235**	202	33	**235**
SCI	349	43	**392**	354	45	**399**
SPO	22	7	**29**	22	7	**29**
SPORT	6	2	**8**	6	2	**8**
STATE-OR-PROVINCE	2503	365	**2868**	2753	401	**3154**
SUBAREA-FACILITY	253	39	**292**	260	39	**299**
TIME	559	50	**609**	564	50	**614**
TOWN	8685	1224	**9909**	13084	1892	**14976**
UNIT	6	1	**7**	50	4	**54**
WATER-BODY	166	35	**201**	186	35	**221**
WEBSITE	1777	284	**2061**	1777	284	**2061**

Table 5: Distribution of 53 NER tags across three hierarchical Levels (parent, sub, and sub-sub) in training and development sets for subtask 1 (flat NER) and subtask 2 nNested NER) in the $\text{Wojood}_{\text{Fine}}$ 2024 dataset.