Thai Nested Named Entity Recognition Corpus

Weerayut Buaphet[†], Can Udomcharoenchaikit[†,*], Peerat Limkonchotiwat[†],
Attapol T. Rutherford[‡], Sarana Nutanong[†]
[†]School of Information Science and Technology, VISTEC, Thailand
[‡]Department of Linguistic, Chulalongkorn University, Thailand
{Weerayut.b_s20, canu_pro, peerat.l_s19,snutanon}@vistec.ac.th,
attapol.t@chula.ac.th

Abstract

This paper presents the first Thai *Nested Named Entity Recognition (N-NER)* dataset. Thai N-NER consists of 264,798 mentions, 104 classes, and a maximum depth of 8 layers obtained from news articles and restaurant reviews, a total of 4894 documents. Our work, to the best of our knowledge, presents the largest non-English N-NER dataset and the first non-English one with fine-grained classes. To understand the new challenges our proposed dataset brings to the field, we conduct an experimental study on (i) cutting edge N-NER models with the state-of-the-art accuracy in English and (ii) baseline methods based on well-known language model architectures. From the experimental results, we obtain two key findings. First, all models produce poor F1 scores in the tail region of the class distribution. There is little or no performance improvement provided by these models with respect to the baseline methods with our Thai dataset. These findings suggest that further investigation is required to make a multilingual N-NER solution that works well across different languages. The dataset and code are available at: github.com/vistec-AI/Thai-NNER.git

1 Introduction

Named Entity Recognition (NER) is a task of extracting named entities from given text. It identifies the span of each entity and categorizes the identified span into an entity category. NER is essential in many downstream tasks, e.g., entity linking, question answering, and knowledge graph. In addition, Yamada et al. (2020) show that the contextualized representations that include entity information can improve many downstream tasks.

The conventional NER paradigm can only label one entity type for each entity span. For example, the entity "Chiang Mai University" will be considered as a single span ignoring the nested structure of the term "Chiang Mai," which is the name

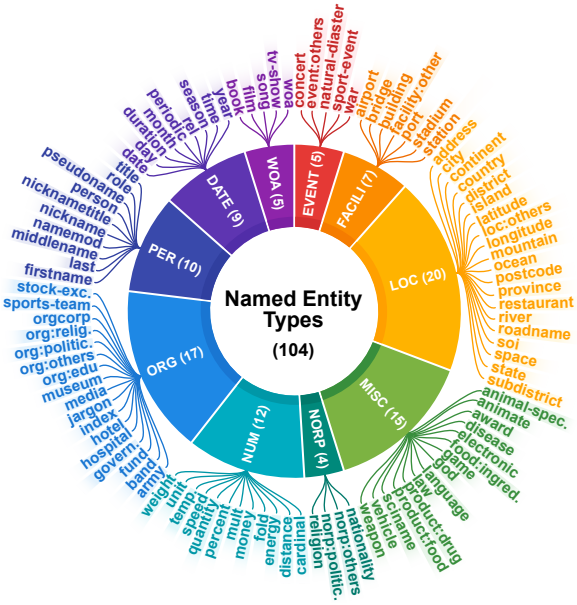


Figure 1: An overview of named-entity classes in Thai N-NER corpus. Our corpus contains 104 fine-grained classes which can be combined into 10 coarse-grained classes. Each box represents a coarse-grained class, and each row within a box represents a fine-grained class.

of the city that the university is situated in. As a result, we may overlook critical information that may have an impact on the language understanding in a downstream task. To mitigate this drawback, one may introduce a *nested* structure into the NER problem. Let us again consider the "Chiang Mai University" example. In addition to annotating the entire span as an organization, *N-NER* also identifies the sub-entity of "Chiang Mai" as a location. This feature can be useful in a downstream task that requires linking an entity to useful references, e.g., a university to its affiliated city.

Considerable research attention has been dedicated to formulating a technique to solve the N-NER problem (Straková et al., 2019a,b; Lin et al., 2019; Wang et al., 2020a; Luo and Zhao, 2020; Shibuya and Hovy, 2020; Wang et al., 2020b). One can use an N-NER model to recursively decompose a complex entity into a tree structure of sub-

* - Corresponding author: canu_pro@vistec.ac.th

entities and have them annotated accordingly.

While N-NER has many potential benefits to downstream tasks that require deep language understanding, there is still a lack of datasets for low-resource languages to help develop reliable N-NER models. In order to train N-NER models, we need a dataset with hierarchical information of each named entity. N-NER datasets are available in several languages. Especially, English, a high resource language, has a few N-NER datasets available for multiple domains (Doddington et al., 2004; Walker et al., 2006; Kim et al., 2003; Ringland et al., 2019) including news, social media, and molecular biology.

The diversity of N-NER corpora is only available in English. N-NER datasets are not as widely available for other languages, let alone the diversity of corpora. In German, another high-resource language, there is only one N-NER dataset available (Benikova et al., 2014). For low-resource languages, such as Vietnamese, the two available datasets (Huyen and Luong, 2016; Nguyen et al., 2018) are still small compared to a large N-NER dataset in English (Ringland et al., 2019).

In this paper, we address the scarcity of non-English N-NER resources by introducing a Thai N-NER dataset. Despite over 58 million internet users[1], the Thai language suffers from the lack of annotated resources to build NLP systems. We propose a *Thai N-NER* dataset comprising 264,798 entity mentions obtained from 4,894 documents. In addition to the nested entity structure, we also have more than one hundred classes providing great fidelity in entity categorization as shown in Figure 1. The number of entity mentions and variety of entity classes are comparable to a large N-NER dataset in English (Ringland et al., 2019). Our dataset contains text samples, in both formal and colloquial settings, from news articles and restaurant reviews. Additionally, our corpus allows for the multilingual evaluation of "language-agnostic" deep learning models, which is the current NLP research trend. To facilitate future N-NER research, we make the dataset, the annotation guideline, and the model weights publicly available.

To summarize, our contributions are as follows:

- We create the first Thai N-NER dataset annotated with extensive tagsets that cover a wide range of use cases.
- We evaluate three recent state-of-the-art (SOTA) N-NER models on our dataset and study the effect of long-tail classes.
- We develop an N-NER benchmark comprising strong baselines for the Thai language that learn each annotation layer separately and achieve performance comparable with the three recent SOTA N-NER models.

[1] https://www.internetworldstats.com/stats3.htm

2 Related Work

In this section, we discuss various attempts on N-NER corpora. As shown in Table 1, existing N-NER corpora are mostly high-resource languages, i.e., English and German, while Vietnamese is the only Asian language that has an N-NER dataset. In terms of the number of classes, it is also worth-noting that three out of six corpora has less than ten classes and only NNE (Ringland et al., 2019) has more than 100 classes. The details of these corpora are given as follows.

Dataset	Docs	Tokens	En.types	Depth	Mentions
English					
NNE	2,312	1.1M	114	6	279,795
GENIA	2,000	0.5M	36	4	92,681
ACE-2005	464	0.3M	7	6	30,966
German					
NoSta-D	-	0.6M	4	2	41,005
Vietnamese					
VLSP-2018	1,282	-	3	2	35,817
Danish					
Dan+	-	0.1M	4	2	6,425
Thai					
Our	4,894	1.2M	104	8	264,798

Table 1: The statistical information comparison between our Thai N-NER corpus and N-NER corpora in other languages. Note that, we obtain the statistical information of NNE, GENIA, and ACE-2005 from Ringland et al. (2019)

ACE-2004 (Doddington et al., 2004) and **ACE-2005** (Walker et al., 2006) are early examples of N-NER datasets. **ACE-2005** (Walker et al., 2006) dataset comprises 30,966 mentions from 12,548 sentences with 7 coarse-grained entity types. In addition to N-NER annotations, ACE-2005 also contains labels for other tasks such as recognition of relation and event extraction.

GENIA (Kim et al., 2003) introduces an N-NER data for bioinformatics. This project provides a high-quality corpus annotated for biological entity names. The dataset is composed of 2,000 abstracts, 92,681 mentions from 9,533 sentences with 32 entity types.

NNE (Ringland et al., 2019) is a recent large fine-grained N-NER dataset composed of 114 classes. Unlike previous N-NER corpora, the NNE dataset annotates entities with more details. For example, "6 September 2019", a date named entity mention, in the NNE dataset, each element in this mention is annotated with finer detail, "6" is annotated with day tag, "September" with month tag, and "2019" with year tag.

NoSta-D (Benikova et al., 2014) is the first and only German N-NER dataset. NoSta-D is composed of 41,005 mentions, 12 entity types, and 31,300 sentences from the German Wikipedia and online news. The previous German NER dataset, CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003), shows that the performance of German is lower than English's[2]. However, the German CoNLL-2003 dataset is known to be inconsistent because it was annotated by non-native speakers. Hence, NoSta-D aims to provide a high quality free public NER dataset by using native speakers as annotators. In contrast to previous N-NER corpora, NoSta-D has a less restrictive copyright license.

VLSP–2018 (Nguyen et al., 2018) is a standard benchmark for Vietnamese N-NER. It was designed for the Vietnamese NER shared task to foster the development of high-quality open-source software. This dataset contains 35,817 mentions from 1,282 documents with 3 entity types.

DAN+ (Plank et al., 2020) presents the first N-NER dataset for Danish. This work investigates the possibility of transfer-learning between languages for the N-NER task. Moreover, DAN+ is a multi-domain dataset; they also study the challenges of domain-shift in their dataset. The dataset contains 6,425 mentions, 130,095 tokens, 4 classes from 6,867 sentences, obtained from multiple domains such as news and social media (Reddit, Twitter, and Arto).

NoSta-D, VLSP–2018, and DAN+ have a modest corpus size and a small number of entity types comparing to the NNE dataset. This shows that there is still a resource gap for non-English corpora. On the other hand, for Thai, there are only coarse-grained flatten-NER datasets which are publicly available (Tirasaroj and Aroonmanakun, 2009; Boonkwan et al., 2020).

3 Thai N-NER corpus

In this section, we introduce Thai N-NER–the first Thai-Nested Named Entity Recognition dataset. Our dataset is comparable to the NNE corpus (Ringland et al., 2019), which is the most elaborate English N-NER dataset in terms of the number of mentions, depth, and the number of classes. In particular, Thai N-NER comprises 264,798 mentions organized into 104 classes and has a maximum depth of 8 layers.

3.1 Data Collection Procedure

To create the dataset, we gather 4,894 documents from two different domains: news articles and restaurant reviews. In particular, we obtain 4,396 news articles from *Prachathai*[3], a news website, and 498 restaurant reviews from *Wongnai*[4], a crowd-sourced restaurant review platform.

The Thai language poses a challenge to the annotation process. Previous work often conducts the annotation at the token level, which is quite convenient for more accurate annotation. However, the lack of clear word boundaries in the Thai writing system does not allow us to easily annotate at the word-level because the data must be word-segmented first, automatically or not. Automatic word segmentation often makes errors around out-of-vocabulary words, which are exactly what we need to annotate. Consequently, the annotation at the word level is not suitable for our purposes if the data are not manually segmented first, which incurs more cost of annotation. Annotating character-level data does not solve the problem either, because annotators are more prone to make an error.

To ease and reduce annotation errors, we provide our annotators with syllable-segmented data instead. Aroonmanakun (2002) shows that syllable segmentation can resolve many word-level ambiguities in Thai. Plus, automatic syllable segmentation can be done at a near-perfect accuracy because the task is mostly solved by orthographic rules, assuming few typos exist in the data (Chormai et al., 2020). With syllable boundary indicators, we can avoid errors from word segmentation. In addition, syllable-segmented data reduces the number of indices drastically, which in turn reduces annotation errors. Appendix A.4 provides

[2]https://www.clips.uantwerpen.be/conll2003/ner/
[3]https://huggingface.co/datasets/prachathai67k
[4]https://github.com/wongnai/wongnai-corpus

Word				Layer 1	Layer 2	Layer 3	Layer 4	Layer 5	Layers 6-8
ประธาน	pràtʰa:n.	president	‘President’	B-Role	O	O	O	O	O
คณะกรรมการ	kʰáná.kammáka:n	committee	‘Committee’	I-Role	B-Org_po.	O	O	O	O
40	sì:sìp	40	‘40’	I-Role	I-Org_po.	B-Event_oth.	B-Duration	S-Cardi.	O
_	_	_	_	I-Role	I-Org_po.	I-Event_oth.	I-Duration	S-Unit	O
ปี	pi:	years	‘Year’	I-Role	I-Org_po.	I-Event_oth.	E-Duration	O	O
_	_	_	_	I-Role	I-Org_po.	I-Event_oth.	O	O	O
14	sìpsì:	14	‘14’	I-Role	I-Org_po.	I-Event_oth.	B-Date	S-Day	O
ตุลา	tùla:	oct	‘October’	I-Role	I-Org_po.	I-Event_oth.	E-Date	S-Month	O
เพื่อ	pʰɯ̂a	for	‘For’	I-Role	I-Org_po.	I-Event_oth.	O	O	O
ประชาธิปไตย	pràtɕʰa:tʰíppàtaj	democracy	‘Democracy’	I-Role	I-Org_po.	I-Event_oth.	S-Norp_po.	O	O
สมบูรณ์	sŏmbu:n	complete	‘Complete’	E-Role	E-Org_po.	E-Event_oth.	O	O	O

Table 2: Our corpus is available in CoNLL format. The first column contains words and other eight columns contain labels in each layer

an example of how syllable-segmented data can help improve annotation experience.

3.2 Annotation Guideline

Inspired by the guideline from (Ringland et al., 2019), we design an N-NER annotation guideline for Thai. To cover a wide range of use cases, our N-NER tagsets comprises coarse-grained and fine-grained categories. While fine-grained categories create extra burden for the annotation and may result in more errors, the trade-off is worth it because finer-grained categories lend themselves to be nested within a coarser category. For example, as shown in Figure 2, พ.ต.อ.ประเวศน์ มูลประมุข (pʰan.tamrùat.ʔèk pràwêːt muːnpràmúk) *‘Police Colonel Prawet Munpramuk’* is tagged with PER–a coarse-grained class which encapsulates other fine-grained classes related to person name. Within a coarse-grained mention, we include nested fine-grained information to each nested named-entity element to give more detail. For example, we annotate พ.ต.อ. (pʰan.tamrùat.ʔèk) *‘Police Colonel’* with title name, ประเวศน์ (pràwêːt) *‘Prawet’* with first name, and มูลประมุข (muːnpràmúk) *‘Munpramuk’* with last name.

Figure 2: An example of a nested named-entity annotation. Entity mentions in a deeper layer must be within the span of the entity mention in the previous layer to give finer details for the coarse-grained.

Apart from the description for each entity class, we provide annotators with case studies for common annotating complications. One frequent complication during the annotation process is ambiguous named entities that change their categories depending on the context. The same string annotated as one category in one context might be annotated as another in a different context. To illustrate this complication, we provide the following example:

(1) ทหาร ไทย โดน จับ
tʰáhăːn tʰaj doːn tɕàp
military Thai is arrested
‘Thai military is arrested’

(2) ทหาร ไทย สั่ง ห้าม ออก จาก บ้าน
tʰáhăːn tʰaj sàŋ hâːm ɔːk tɕàːk bâːn
military Thai ordered prohibit leave from house
‘Thai military prohibits going outside of the house’

In the example above, the word ทหารไทย (tʰáhăːn tʰaj) *‘Thai military’* is not always a named entity depending on the context. In example sentence (1) ทหารไทยโดนจับ (tʰáhăːn tʰaj doːn tɕàp) *‘Thai military is arrested’*, *‘Thai military’* is not a named entity because *‘Thai military’* refers to a Thai soldier. In contrast, the example sentence (2) ทหารไทยสั่งห้ามออกจากบ้าน (tʰáhăːn tʰaj sàŋ hâːm ɔːk tɕàːk bâːn) *‘Thai military prohibits going outside of the house’*, *‘Thai military’* is a named entity because it refers to the Thai military institution.

A named entity mention that is composed of nested named entities can be regarded as a tree structure. Specifically, the first level of a mention is the outermost or the largest entity span of the mention. The nested entities within the mention in each level must not overlap and cannot span outside of the mention. We provide

an example of an issue that arises from overlapping annotations in Appendix A.3. Each coarse-grained entity type can appear in any level of the nested structure. However, fine-grained entity type must be nested under its coarse-grained entity type. As shown in Table 2, ประธานคณะกรรมการ 40 ปี 14 ตุลาเพื่อประชาธิปไตยสมบูรณ์ (pràtʰa:n.kʰáná.kammáka:n sì:sìp pi: sìpsì: tùla: pʰûa pràtɕʰa:tʰíppàtaj sǒmbu:n) *'The 40-year 14 Oct for complete democracy committee president'* is the first level of a named-entity mention which is annotated as a role type and the nested structure also contained other coarse-grained mentions such as date or duration. However, fine-grained entity mentions, such as day and month, can only be nested inside the date class.

3.3 Annotation Quality Control Procedure

To make our dataset reliable, we require that annotators have a background in linguistics and are properly trained to annotate under our guidelines. We also do quality control and evaluation to verify the quality of our dataset.

3.3.1 Annotators

The dataset is manually annotated by 47 linguistically trained annotators. The annotators have the necessary linguistic background and have passed the N-NER guideline understanding test. We provide a communication channel to discuss annotation issues among the annotators and the project manager. We use Datasaur.ai[5] platform for the annotators to label the data according to our guideline, using syllable span highlighting to designate each span as a specific entity.

3.4 Annotation Verification Process

Firstly, we manually check the quality of annotated randomly data to find common mistakes. To find more annotation errors, we extract only the first layer to train a simple flatten CRF model. Then we use the CRF model to filter its prediction errors for further error analysis. Combining the errors found by both humans and the model, we conduct an error analysis to find the pattern of mistakes from annotators. Frequent annotation mistakes are inconsistency tagging, incorrect tagging, and failure to follow the guideline. Then we compile a list of annotation errors and send it back to the annotators to reassess. After the first update, we use a rule-based program to filter overlapping annotations, which violate our guideline, then list all the documents with overlapping annotations. Moreover, we employ a gazetteer to filter mislabeled entities. Later, we report the list of overlapping documents and the list of mislabeled entities to the annotators to correct all the annotation errors.

After the second update, to inspect our dataset quality, we train an N-NER model from Shibuya and Hovy (2020) to see whether our data can be used to train the model and to filter out more annotation errors. The test score is 75.44% F1 score. We then use the model's prediction errors to filter out more annotation mistakes and report them to the annotators for another correction session.

Then, we split our dataset into 80% for a training set and 20% for a test set, then re-annotate the test set with two annotators to validate. Finally, the third annotator correct the annotation mismatches between the first two annotators.

We use the Cohen's Kappa agreement score to benchmark the reliability of our dataset. We compute the inter-annotator agreement using eight sampled documents composed of 2,922 tokens. We calculate the Cohen's Kappa agreement score using two labeling schemes: CoNLL and Pyramidal, see Appendix A.5 for further descriptions. The agreement scores are given as follows:

- CoNLL: 0.79;
- Pyramidal: 0.85;

These high agreement scores imply that our dataset is of good quality.

3.5 Data Format

To make our dataset convenient for research usage, we provide our dataset in CoNLL-format as shown in Table 2. We define the word boundaries in the dataset by using a maximal matching tokenizer from PyThaiNLP (Phatthiyaphaibun et al., 2016). In addition, we employ the BIOES tagging scheme to indicate the boundary of each named entity mention. Furthermore, we replace each empty space token with "_" in order to keep the integrity of the original text when we convert the CoNLL version back to the original text with no tokenization.

4 Data Statistics

This section discusses the dataset statistics and analyzes the distribution of classes in the dataset. Table 3 shows the dataset statistics of the Thai N-NER. The Thai N-NER corpus contains 1,272,381

[5] https://datasaur.ai

tokens from 4,894 documents. The dataset has 264,798 named entity mentions, 104 entity types, and 8 maximum depth.

Items	Train	Dev	Test	Total
Documents	2,935	979	980	4,894
Tokens	763,421	256,553	252,407	1,272,381
Entity types	104	101	104	104
Max. depth	7	8	6	8
Mentions	155,353	50,501	58,944	264,798
Layer 1	74,281	24,373	26,526	125,180
Layer 2	70,967	23,000	26,942	120,909
Layer 3	8,987	2,799	4,714	16,500
Layer 4	964	284	673	1,921
Layer 5	129	41	82	252
Layer 6	24	1	7	32
Layer 7	1	2	0	3
Layer 8	0	1	0	1

Table 3: The data statistics and distribution of entities in each layer.

The Thai N-NER dataset contains a nested structure for each named-entity mention. The first three layers contain 125,180, 120,909, and 16,500 mentions accounted for 99.2% and mentions all other levels contain 2,209 mentions combined accounted for only 0.8%. The 125,180 first-layer mentions can be divided into 67,168 nested mentions and 58,012 non-nested mentions. We split our dataset into training set, development set, and test set with proportion of 60%, 20%, and 20% respectively. The test set contains all the 104 classes appeared in the training set.

We compare our dataset with other N-NER datasets in other langauges. Table 1 shows the statistics of N-NER datasets between NNE, GENIA, ACE-2005 (English), VLSP-2018 (Vietnamese), Dan+ (Danish), and our dataset (Thai). It should be noted that our dataset is comparable to the existing N-NER datasets in term of the number of tokens and the number of entity types.

One of the challenges in this dataset is class imbalance. Due to the number of classes, the scarcity of data for rare classes contribute to the severity of class imbalance. We visualize the distribution of classes in training set in Figure 3. The graph shows the distribution of mentions per class in training set sorted by frequency. To analyse the severity of class imbalance, we divided the classes into three groups follow Pareto principle: head, body and tail with samples per classes are 80%, 15% and 5% respectively. More precisely, in body and tail parts, they contain only 20% of samples in training set, but consist of 84 classes from 104 classes.

In conclusion, we introduce a dataset for Thai N-NER that is comparable to the standard N-NER dataset in English. Additionally, we point out a challenging long-tail distribution problem in N-NER that allows researchers to explore.

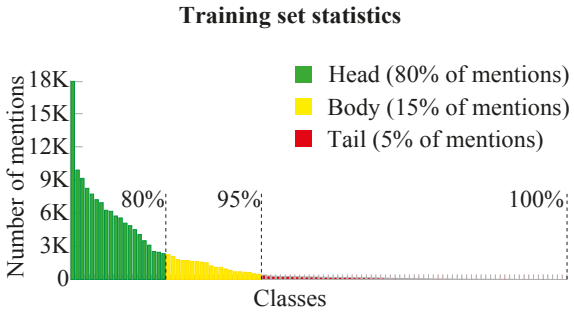


Figure 3: The distribution of classes sorted by frequency shows that rarer classes consist of more than 20% of all instances.

5 Experimental Settings and Results

The objectives of the experimental studies are as follows: the first objective is to help researchers understand how existing techniques perform on our dataset and to help them choose the most appropriate baseline for future research. The second objective is concerned with the distribution of classes which follows the 80-20 Pareto principle. As shown in Figure 3, the top 20% most frequent classes account for 80% of the mentions. We also study how these techniques perform differently at the head, body, and tail parts of the distribution. The third objective is to compare how existing models perform on our Thai dataset with respect to results from existing studies conducted on English datasets.

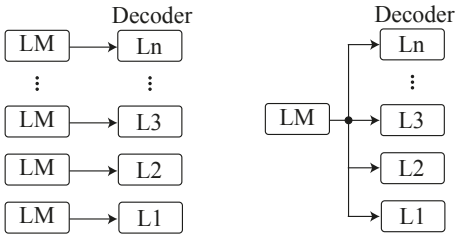
5.1 Comparative N-NER Models

Since there is no existing Thai N-NER model, we formulate comparative solutions based on three approaches. The first approach is to build a baseline N-NER method from a classical machine learning technique. The second approach is applying a Thai language model to perform a span classification task. The third approach is to adapt existing N-NER methods to Thai by replacing their encoders with a Thai language model. For ease of comparison, we apply the best existing Thai language model called WangchanBERTa (Lowphansirikul et al., 2021) to second and third approaches.

Classical ML baseline: CRF model (Minh, 2018) We train multiple CRF models, each model is dedicated to each layer. Then, we merge the pre-

diction results from all layers to form the final N-NER result. For this model, we use the IOB tagging scheme because our dataset has a large number of classes; hence the IOBES scheme will take longer to train.

Deep learning baseline: WangchanBERTa and XLM-RoBERTa. We finetune language model (LM) encoders on our corpus with two architectural variants, LM-separate and LM-shared as shown in Figure 4a and 4b, respectively. For both model, we simply use a fully-connected linear layer as a decoder. For *separate-weight (sp)* version, we assign one encoder-decoder model for each layer. For *shared-weight (sh)* version, we use multiple decoders, one for each layer, while sharing the same encoder. We provide more information about parameter settings in Appendix A.1.



(a) LM-separate (sp). (b) LM-shared (sh).

Figure 4: (a) LM–separate: each level in the nested structure has a full encoder-decoder model trained to predict tags in that specific level only. (b) LM–shared: each nested level has its own dedicated decoder, while sharing the same encoder. L_1 to L_n denotes the depth level.

To compare the performances between monolingual and multilingual BERT variants, we run experiments on both WangchanBERTa (Thai) and XLM-RoBERTa (multilingual).

State-of-the-art Models: We select three recent SOTA N-NER models with open-source accesses and train them on our corpus. To get these models to work for Thai, we replace their encoders with the same Thai language model as the deep learning baselines (Lowphansirikul et al., 2021). For parameter configurations, we use GENIA's parameter configurations to make it possible to do sanity check by reproducing previous results on GENIA

Second-best-learning (Shibuya and Hovy, 2020): This model learns to recursively decode the nested named entities from the outer to the inner nested entities. It is commonly used as a baseline in recent N-NER research. It has strong results for English N-NER.

Pyramid (Wang et al., 2020a): This model learns hierarchical representation from multiple nested levels by using pyramid and inverse pyramid mechanisms. This model currently has the highest score on the NNE dataset.

Locate and Label (Shen et al., 2021): This model divides entity detection into two stages: (i) it locates the entity spans; (ii) it assigns a label to each entity span. It is the most recent state-of-the-art model, it has top-performing scores on ACE-2004 and GENIA corpora.

5.2 Evaluation Settings

We follow the evaluation methodology from (Shibuya and Hovy, 2020), they consider a prediction as a true positive if both the predicted entity span and type are correct. In order to examine the long-tail issue as mentioned in Section 4, we evaluated the effect of long-tail distribution by dividing classes into three groups: head, body, and tail.

5.3 Thai N-NER Results

Table 4 shows the results on different parts of the long-tailed distribution, as well as the overall results on our dataset. Among the three existing SOTA models, the Second-best-learning model has the highest overall performance. It obtains higher F1 scores on the head and body parts of the long-tail distribution, while the Pyramid model obtains the highest F1 score on the tail part.

Interestingly, the deep learning baseline models, WangchanBERTa and XLM-R, can perform on par with all the current SOTA models. As shown in Table 4, the performances of WangchanBERTa models on the body and tail parts, and XLM-R models on the tail part are superior to the best SOTA model.

By having better performances on body and tail parts, while maintaining a high performance on the head part, both of the deep learning baseline models can obtain competitive results compared to all the SOTA models on our corpus.

The performances of models based on the multilingual encoder (XLM-R) are superior to Pyramid and Locate and label models. However, compared to the monolingual encoder (WangchanBERTa), XLM-R models' performances are better than the monolingual models. This suggests the possibility of cross-lingual N-NER tasks. (e.g. transferring cultural-specific named-entity knowledge from English to Thai).

	Models	Head			Body			Tail			All		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Baseline	CRF model	86.06	66.46	75.00	78.30	44.88	57.06	74.07	29.46	42.15	84.60	60.59	70.61
	WangchanBERTa–sp	**90.70**	77.66	83.67	**81.55**	55.90	**66.33**	78.02	26.09	39.10	**89.04**	70.89	78.94
	WangchanBERTa–sh	90.51	79.24	84.50	81.37	55.09	65.70	78.33	30.79	44.20	88.87	72.25	79.70
	XLM-R–sp	90.27	77.39	83.34	80.45	52.71	63.69	75.42	33.04	45.95	88.42	70.56	78.48
	XLM-R–sh	89.45	79.72	84.31	77.29	**58.06**	66.31	71.80	**39.73**	**51.16**	86.93	**73.66**	**79.75**
SOTA	Second-best-learning	87.57	**81.78**	**84.58**	80.12	54.85	65.12	**79.05**	19.41	31.16	86.41	73.49	79.43
	Pyramid	87.59	80.33	83.81	76.07	53.72	62.97	74.20	23.92	36.18	85.65	72.45	78.50
	Locate and label	77.60	80.38	78.97	64.42	56.21	60.04	77.43	18.86	30.33	75.57	72.61	74.06

Table 4: Experimental results nested-NER models divided into head, body, tail, and overall in our dataset

The long-tailed distribution of classes poses a challenge for the N-NER task. The performances across all models quickly deteriorate as we move from the head part of the long-tailed distribution, which represents common classes, to the tail part, which represents infrequent classes. Additionally, there are gaps between precision and recall for all models. These gaps imply that all models have a tendency to generate false negatives more than false positives. We can also see that the precision-recall gap has a tendency to increase as we move from the head to the tail part of the distribution. This result suggests that in order to improve the overall performance, we should pay attention to recall.

In addition, comparing to the results on English N-NER corpora, there is a performance gap for the Thai language. For example, the F1 score of the Pyramid model on the NNE corpus is 94.68, while its performance on our corpus is only 78.50. For the full comparison, see Appendix A.6.

6 Error Analysis

To understand the limitation of current N-NER solutions, we investigate reoccurring mistake patterns from the WangchanBERTa-sp models used in the experimental studies. We categorize the common prediction mistakes into four groups as follows: (1) *Incorrect span prediction*: out of 5,334 prediction errors, 3,103 errors are from span length mismatch as shown in Figure 5. (2) *Ambiguous entity mentions*: mentions with higher class distribution entropy have more error rates. (3) *Ambiguity between fine-grained classes*: there are 1,160 fewer errors when evaluated with coarse-grained ground truths. (4) *Scarcity of training samples*: the model only made 1,380 prediction attempts for mentions in tail classes. While 1,081 of the predictions are correct, there are 3,511 ground truths. The previous section also reveals this issue via the poor recall scores in the tail part of the long-tail distribution. We provide the description of each error pattern along with examples in Appendix A.8.

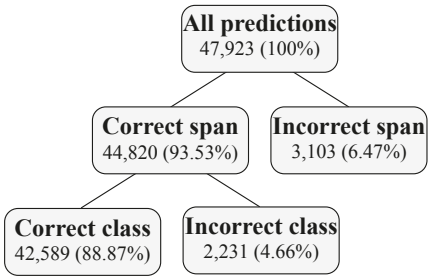


Figure 5: Tree diagram of mention predictions: this tree diagram breaks down predictions from the WangchanBERTa–sh model. It illustrates that a large chunk of prediction errors is from incorrect span predictions.

7 Summary

We present the first Thai N-NER corpus with 104 classes. It has 1,272,381 words, and 264,798 mentions. The size of our corpus is comparable to one of the large N-NER corpora in English. Unlike other Thai NER corpora, in addition to nested structure information, our dataset is annotated with fine-grained entity types to provide more detail of the named entities. This corpus addresses the data scarcity issue for Thai NLP. In addition, it allows NLP researchers to benchmark their methods in a multilingual setting. Moreover, this dataset allows researchers to explore the effect of long-tail distribution. We hope that our dataset will encourage researchers to include Thai in their benchmark and reduce the disparity between Thai and high resource languages.

Ethical Consideration

Our dataset consists of raw text data from two publicly available corpora: Prachatai-67k and Wongnai review. These corpora use public copyright licenses (LGPL and Creative Commons) that en-

able free distribution. The data has a minimal risk for privacy violation since all the data were published in a public space, such as a news site and a restaurant review site. All the news articles and restaurant reviews are meant to be shared publicly, not privately. Hence, the dataset does not contain any confidential information. Our preprocessing step, which includes cleaning data and tokenization, does not alter the original contents of the texts. On average, the annotators were compensated at least twice the local minimum wage. The annotators were paid by the number of entity-mentions annotated and the number of documents that they have read. We distributed the same amount of documents for each annotator for fair consideration. This dataset addresses the data scarcity issue for Thai, which can be considered as a lower-resource language. However, this dataset only includes the central Thai dialect, which most Thai understand. It is also the dialect for official usage and is often used as a written language by Thai internet users. It reduces the language technology disparity gap between Thai and high-resource languages. In addition, it can facilitate researchers and the NLP community to investigate the N-NER task in a multilingual setting. We will open-source the dataset and distribute it publicly under the CC by SA 3.0 license. We will also publish the source code and the models' weights from our experiments to assist the NLP community in N-NER research and reduce unnecessary energy usage from training the models.

Acknowledgments

We would like to thank team members from VISTEC-depa Thailand Research Institute and Hope Data Annotation. The Thai N-NER corpus is supported in part by the Digital Economy Promotion Agency (depa) Digital Infrastructure Fund MP-62-003 and Siam Commercial Bank. This dataset is released as `scb-nner-th-2022`.

We also thank reviewers for their thoughtful reviews and suggestions.

References

Wirote Aroonmanakun. 2002. Collocation and thai word segmentation. *Proc. SNLP and Oriental COCOSDA Workshop, 2002*, pages 68–75.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. NoSta-D named entity annotation for German: Guidelines and dataset. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2524–2531, Reykjavik, Iceland. European Language Resources Association (ELRA).

Prachya Boonkwan, Vorapon Luantangsrisuk, Sitthaa Phaholphinyo, Kanyanat Kriengket, Dhanon Leenoi, Charun Phrombut, Monthika Boriboon, Krit Kosawat, and Thepchai Supnithi. 2020. The annotation guideline of lst20 corpus. *arXiv preprint arXiv:2008.05055*.

Pattarawat Chormai, Ponrawee Prasertsom, Jin Cheevaprawatdomrong, and Attapol Rutherford. 2020. Syllable-based neural Thai word segmentation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4619–4637, Barcelona, Spain (Online). International Committee on 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.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Nguyen Thi Minh Huyen and Vu Xuan Luong. 2016. VLSP 2016 shared task: Named entity recognition. *Proceedings of Vietnamese Speech and Language Processing (VLSP)*.

J-D Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2003. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl_1):i180–i182.

Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2019. Sequence-to-nuggets: Nested entity mention detection via anchor-region networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5182–5192, Florence, Italy. Association for Computational Linguistics.

Lalita Lowphansirikul, Charin Polpanumas, Nawat Jantrakulchai, and Sarana Nutanong. 2021. Wangchanberta: Pretraining transformer-based thai language models. *arXiv e-prints*, pages arXiv–2101.

Ying Luo and Hai Zhao. 2020. Bipartite flat-graph network for nested named entity recognition. In *Proceedings of the 58th Annual Meeting of the Asso-*

ciation for Computational Linguistics, pages 6408–6418, Online. Association for Computational Linguistics.

Pham Quang Nhat Minh. 2018. A feature-based model for nested named-entity recognition at vlsp-2018 ner evaluation campaign. *Journal of Computer Science and Cybernetics*, 34.

Huyen TM Nguyen, Quyen T Ngo, Luong X Vu, Vu M Tran, and Hien TT Nguyen. 2018. Vlsp shared task: Named entity recognition. *Journal of Computer Science and Cybernetics*, 34(4):283–294.

Wannaphong Phatthiyaphaibun, Korakot Chaovavanich, Charin Polpanumas, Suriyawongkul Arthit, Lalita Lowphansirikul, and Pattarawat Chormai. 2016. PyThaiNLP: Thai Natural Language Processing in Python.

Barbara Plank, Kristian Nørgaard Jensen, and Rob van der Goot. 2020. DaN+: Danish nested named entities and lexical normalization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6649–6662, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Charin Polpanumas and Wannaphong Phatthiyaphaibun. 2021. thai2fit: Thai language implementation of ulmfit.

Nicky Ringland, Xiang Dai, Ben Hachey, Sarvnaz Karimi, Cecile Paris, and James R. Curran. 2019. NNE: A dataset for nested named entity recognition in English newswire. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5176–5181, Florence, Italy. Association for Computational Linguistics.

Yongliang Shen, Xinyin Ma, Zeqi Tan, Shuai Zhang, Wen Wang, and Weiming Lu. 2021. Locate and label: A two-stage identifier for nested named entity recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2782–2794, Online. Association for Computational Linguistics.

Takashi Shibuya and Eduard Hovy. 2020. Nested named entity recognition via second-best sequence learning and decoding. *Transactions of the Association for Computational Linguistics*, 8:605–620.

Jana Straková, Milan Straka, and Jan Hajic. 2019a. Neural architectures for nested ner through linearization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5326–5331.

Jana Straková, Milan Straka, and Jan Hajic. 2019b. Neural architectures for nested NER through linearization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5326–5331, Florence, Italy. Association for Computational Linguistics.

Nutcha Tirasaroj and Wirote Aroonmanakun. 2009. Thai named entity recognition based on conditional random fields. In *2009 Eighth International Symposium on Natural Language Processing*, pages 216–220.

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.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. ACE 2005 Multilingual Training Corpus LDC2006T06. `https://catalog.ldc.upenn.edu/LDC2006T06`. Linguistic Data Consortium.

Jue Wang, Lidan Shou, Ke Chen, and Gang Chen. 2020a. Pyramid: A layered model for nested named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5918–5928, Online. Association for Computational Linguistics.

Yu Wang, Yun Li, Hanghang Tong, and Ziye Zhu. 2020b. HIT: Nested named entity recognition via head-tail pair and token interaction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6027–6036, Online. Association for Computational Linguistics.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

A Appendix

A.1 Parameter Settings

For all the deep learning baselines, we use the following parameter configuration: We employ Adam optimizer with a learning rate of 1e-5. We utilize a learning rate decay scheduler that reduces the learning rate every 50 epochs by multiplying the decay factor of 0.1. The maximum training epoch is 500, and we early stop if there is no improvement for 16 epochs. We use the last checkpoint for WanchanBERTa-sh and XLM-R-sh to evaluate, but for the WanchanBERTa-sp and XLM-R-sp, we use the epoch that model does not further improve when training.

For the Locate and Label model, we made further modifications to the model to use it for the Thai language. Unlike the original work, the sequence length limitation of WangchanBERTa is lower than BERT–large version (Devlin et al., 2019), we use only ten words from each neighboring sentence as the context words to keep the input sequence length within the limitation. In addition, apart from contextualized word embeddings, Locate and Labels also includes static word embeddings–GloVE. We replace the GloVE word embeddings with the static word embeddings layer of thai2fit (Polpanumas and Phatthiyaphaibun, 2021). thai2fit was trained on wisesight-sentiment [6], prachathai-64k [7], and TH-wikipedia [8].

A.2 Coarse-grained vs Fine-grained Scores

Table 5 compares the WangchanBERTa-sh model's performances between the coarse-grained and fine-grained ground truths. We converted fine-grained labels to their respective coarse-grained labels to examine the negative effect from the ambiguity between fine-grained classes. Table 5 shows that there is a small gap between coarse-grained and fine-grained evaluations. It suggests that adding fine-grained information to the dataset does not introduce a major challenge for N-NER models. Nevertheless, errors from ambiguity between fine-grained classes still constitute a considerable amount of models' prediction errors.

[6] https://github.com/PyThaiNLP/wisesight-sentiment
[7] https://github.com/PyThaiNLP/prachathai-67k
[8] https://dumps.wikimedia.org/thwiki

Classes	Coarse-grained P	Coarse-grained R	Coarse-grained F1	Fine-grained P	Fine-grained R	Fine-grained F1
PER	93.14	77.65	84.69	91.06	75.95	82.82
LOC	90.82	74.45	81.83	88.17	72.29	79.44
DATE	96.12	87.50	91.61	95.98	87.37	91.47
ORG	84.07	60.03	70.04	76.04	54.24	63.32
NORP	77.85	34.98	48.27	74.26	33.35	46.03
FACILI.	64.69	37.64	47.59	60.62	35.27	44.60
EVENT	48.03	20.33	28.57	45.52	19.27	27.08
WOA	69.62	19.64	30.64	59.49	16.79	26.18
MISC	83.24	41.66	55.53	81.98	41.03	54.69
NUM	94.66	89.50	92.01	93.68	88.57	91.05
TOTAL	91.30	74.24	81.89	88.87	72.25	79.70

Table 5: Fine-grained and coarse-grained evaluations of the WangchanBERTa-sh model

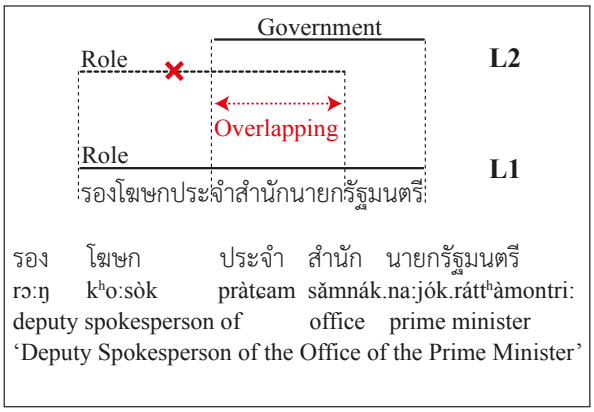


Figure 6: The annotation scheme does not allow overlapping between entities in the same layer.

A.3 Issue with Overlapped Annotations

Similar to a morphological parse tree, a nested entity annotation structure does not allow overlapping between entities in the same depth level. For example, in Figure 6, รองโฆษกประจำสำนักนายกรัฐมนตรี (rɔ:ŋ kʰo:sòk pràtɕam sămnák.na:jók.rátthàmontri:) *'Deputy Spokesperson of the Office of the Prime Minister'* is the first level of the nested named entity mention.

In the second layer, we do not allow an annotator to tag รองโฆษกประจำสำนักนายก (rɔ:ŋ kʰo:sòk pràtɕam sămnák.na:jók) *'Deputy Spokesperson of the Office of the PM'* with a role tag and สำนักนายกรัฐมนตรี (sămnák.na:jók.rátthàmontri:) *'Office of the Prime Minister'* with a government tag, because it creates two chunks that share the word สำนักนายก (sămnák.na:jók) which is a abbreviated form of *'Office of the Prime Minister'*. This would violate the tree structure. In addition, annotating รองโฆษกประจำสำนักนายก (rɔ:ŋ kʰo:sòk pràtɕam sămnák.na:jók) *'Deputy Spokesperson of the Office of the PM'* as an instance of named entity suggests that *'Deputy Spokesperson of the Office of the PM'*

Models	ACE-2004			ACE-2005			GENIA			NNE		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Second-best-learning	85.23	84.72	84.97	83.30	84.69	83.99	77.46	76.65	77.05	-	-	-
Pyramid	87.71	87.78	87.74	85.30	87.40	86.34	-	-	-	94.30	95.07	94.68
Locate and label	87.44	87.38	87.41	86.09	87.17	86.67	80.19	80.89	80.54	-	-	-

Table 6: The performances of the recent SOTA N-NER models on English datasets, we include the performances from their original papers.

is a noun phrase and รัฐมนตรี (rátthàmontri:) *'Minister'* is a noun modifier, which is semantically incorrect. As far as compositional semantics is considered, the nested structure of named entities should not contain overlapping entities in the same level.

A.4 Annotation Experience Improvement with Syllable Segmentation

Syllable segmentation enhances the annotation experience since there are fewer choices than selecting named-entity boundaries at character-level. In addition, Thai syllable segmentation also has a near-perfect accuracy which makes it more suitable word segmentation.

For example, given an input text นายมะกะตา อาแวเซะ (Mr.Makata Arvasa), the syllable-segmented output is นา|ย|มะ|กะ|ตา| |อา|แว|เซะ and the word-segmented output from a standard word tokenizer is นา|ยมะ|กะ|ตา| |อา|แวเซะ.

Since นาย is a title word, we can find the NE's span from syllable tokens นา|ย → นาย. However, we cannot recover from a word segmentation error นา|ยมะ ↛ นาย.

As for the dataset, we present the word-level version because it is a common preprocess technique. We combine the results from word segmentation with the NE boundaries from annotators to ensure that the boundaries for NEs are guaranteed to be correct.

A.5 Annotation Verification Process

CoNLL: we format our dataset according to the CoNLL schema, then calculate the Cohen's Kappa by comparing agreements of annotated entities layer by layer. The CoNLL schema takes the mention's token length into account. For each disagreed mention, we count each disagreed token as one disagreement. Therefore, mentions with more token length may have more disagreement counts. In addition, if there is a mismatch within the same layer, we count it as a disagreement even though the annotations might agree if we were to compare them from different layers.

Pyramidal: we format the labels in a pyramidal manner, where we generate all possible n-gram entity span candidates for each text sequence and assign them to layers according to their lengths in the same fashion as the Pyramid model (Wang et al., 2020a). Then we compare agreements of annotated candidates between the two annotated data. We calculated the score on both character level and token level, and found no difference. We report the score on the token level. Pyramidal scheme counts each disagreed mention as one disagreement despite its length. Since Thai has no word boundary, the pyramid scheme always provides a consistent score despite using it on a different word segmentation that varies the token length.

A.6 The Performances of the Recent SOTA N-NER Models on English Datasets

This study compares the performances of the N-NER models between Thai and English N-NER datasets. Table 4 shows the results on the Thai N-NER dataset, and Table 6 shows the results on English N-NER datasets. We can see that, when compared to the English results, all N-NER models performed poorer on the Thai dataset. For example, the F1-score of Pyramid on the NNE dataset (the most similar dataset compared to our work) is 94.68%, while the overall F1-score of Pyramid for Thai N-NER is only 78.50%. Although both datasets are similar in size, design, and diversity of entity classes, the performance gap is 16.18%. Experimental results verify that there is a performance gap between Thai and English N-NER.

Furthermore, some model is based on the BERT-large model, but Thai has only one BERT-based pretrained model which is based on RoBERTa (WangchanBERTa). This may have a direct affect on the performance gap. For example, the Locate and Label is based on the BERT-large model; replacing BERT-large with WangchanBERTa can effect the performance directly. Despite having the best performances across multiple English N-NER datasets, Locate and Label has the lowest score on the Thai N-NER dataset when compared to other SOTA models.

A.7 Mention Distribution

Table 7 shows the mention frequency of each fine-grained entity type in our corpus before the train-test split. For each nested structure, we count all annotated mentions, not just the outermost mention. This table reveals classes with extremely low frequency which contribute to poor performances on the tail part of the long-tailed distribution.

A.8 Error Analysis

Incorrect span prediction: mismatches between the length of the predicted spans and the ground truths contribute to a large chunk of prediction errors.

Figure 5 shows that out of 47,923 predicted mentions, 5,334 are incorrect. 3,103 out of 5,334 incorrect predicted mentions are due to the fact that the positions of the predicted spans are not correctly aligned with the positions of the ground truths. Often, we can find this error in the predictions for entity mentions that are very long. For example, consider the following text segment:

(1) อาคาร รัฐประศาสนภักดี ชั้น 6
ʔa:ka:n. rátthàpràsà:tsànáphákdi: tɕʰán hòk
building Ratthaprasatsanaphakdi floor 6
ถนน แจ้งวัฒนะ แขวง
tʰànǒn. tɕɛ̂ːŋ.wátthánáˊ kʰwɛ̌ːŋ.
road chaengwatthana subdistrict
ทุ่งสองห้อง เขต หลักสี่
tʰûŋsɔ̌ːŋhɔ̂ŋ kʰèːt. làk.sìː
thungsonghong district laksi
กรุงเทพมหานคร
kruŋtʰêːp.máhǎː.ná.kɔːn
Bangkok
10210
nùŋ.sǔː.n.sɔ̌ːŋ.nùŋ.sǔː.n
10210
'Ratthaprasasanabhakdi Building, 6th Floor, Chaeng Watthana Road, Thung Song Hong Subdistrict, Lak Si District, Bangkok 10210'

This large text segment is just one entity span for the address class. If a N-NER model yields a predicted span that does not cover the whole text segment, even by just one word, then we consider the prediction as incorrect.

Ambiguous entity mentions: models may fail to disambiguate entity mentions that can belong to different classes depending on the context. For example, "English" can be tagged with different classes such as Language, National, or Location depending on the context.

We use normalized class distribution entropy to quantify the effect of ambiguous entity mentions. We investigate entity mentions that can appear as different classes in the training set and calculate their entropy according to their class distribution in the training set. Then we measure the error rates of these mentions in the test set. We split entity mentions into three bins according to their entropy values: $[0, 0.33)$, $[0.33, 0.66)$,$[0.66, 1.0]$. We found that the average error rates of the three bins are as follows: 23.43%, 37.07%, and 69.28%, respectively. This confirms that ambiguity of entity mentions has a deleterious effect on the N-NER model.

Ambiguity between fine-grained classes: there are fine-grained classes that have subtle differences in meaning between them and often appear in similar contexts. For example, the *government* tag refers to governmental organizations such as, government departments, while *org:political* refers to political organizations, such as political parties and advocacy groups.

As mentioned in Appendix A.2, using coarse-grained ground truths to evaluate can reveal the detrimental effect of ambiguity between fine-grained classes. There are 1,160 mentions that would be predicted correctly, if we were to use coarse-grained ground truths instead.

Scarcity of training samples: there are some classes that models do not give any prediction because the number of training samples is too low, for example, *food:ingredient*, *vehicle*, *org:religious*, *periodic*, and *station*. As a result, all models have a tendency to generate false negatives more than false positives. This is the issue we mentioned in Section 5.3. Moreover, the best Thai N-NER model, WangchanBERTa-sh, tends to produce more predictions for the head classes, accounting for 80% of the mention distribution, than body and tail classes.

Classes	Counts	%	Classes	Counts	%	Classes	Counts	%
cardinal	30457	11.502	norp:others	1057	0.399	airport	166	0.063
person	16358	6.178	army	1051	0.397	song	146	0.055
firstname	14896	5.625	percent	1026	0.387	middlename	134	0.051
unit	14069	5.313	disease	826	0.312	mountain	126	0.048
government	13763	5.198	product:food	678	0.256	namemod	123	0.046
country	11979	4.524	religion	675	0.255	station	115	0.043
title	11766	4.443	nickname	625	0.236	award	111	0.042
role	11366	4.292	language	607	0.229	film	106	0.040
last	10315	3.895	state	591	0.223	weight	102	0.039
month	9602	3.626	book	539	0.204	ocean	89	0.034
province	9141	3.452	restaurant	503	0.190	port	78	0.029
day	8585	3.242	continent	480	0.181	energy	74	0.028
date	8096	3.057	fund	414	0.156	space	67	0.025
year	7569	2.858	river	413	0.156	product:drug	64	0.024
quantity	7064	2.668	address	405	0.153	animate	62	0.023
org:political	5796	2.189	pseudoname	402	0.152	sports-event	51	0.019
media	5560	2.100	weapon	402	0.152	fold	49	0.019
org:other	4449	1.680	hospital	391	0.148	woa	48	0.018
loc:others	4200	1.586	electronics	376	0.142	stadium	45	0.017
facility:others	3852	1.455	jargon	347	0.131	sports-team	44	0.017
district	3800	1.435	natural-disaster	346	0.131	band	42	0.016
org:edu	3697	1.396	distance	331	0.125	season	37	0.014
duration	3230	1.220	building	302	0.114	war	37	0.014
law	3144	1.187	island	298	0.113	museum	37	0.014
orgcorp	2929	1.106	animal-species	291	0.110	stock-exchange	36	0.014
rel	2920	1.103	sciname	290	0.110	god	31	0.012
nationality	2876	1.086	food:ingredient	281	0.106	game	24	0.009
norp:political	2682	1.013	tv-show	257	0.097	postcode	17	0.006
time	2643	0.998	vehicle	243	0.092	temperature	11	0.004
money	2055	0.776	hotel	210	0.079	longtitude	8	0.003
city	1867	0.705	nicknametitle	209	0.079	latitude	7	0.003
event:others	1853	0.700	periodic	204	0.077	index	5	0.002
subdistrict	1738	0.656	org:religious	204	0.077	speed	5	0.002
mult	1542	0.582	soi	200	0.076	concert	2	0.001
roadname	1195	0.451	bridge	171	0.065	**Total**	**264,798**	

Table 7: The distribution of entity types in our corpus along with their frequency.