




A large English–Thai parallel corpus from the web and machine-generated text

Lalita Lowphansirikul¹ · Charin Polpanumas² ·
Attapol T. Rutherford^{3,4}  · Sarana Nutanong¹

Accepted: 2 March 2021 / Published online: 30 March 2021
© The Author(s), under exclusive licence to Springer Nature B.V. 2021

Abstract The primary objective of our work is to build a large-scale English–Thai dataset for training neural machine translation models. We construct scb-mt-en-th-2020, an English–Thai machine translation dataset with over 1 million segment pairs, curated from various sources: news, Wikipedia articles, SMS messages, task-based dialogs, web-crawled data, government documents, and text artificially generated by a pretrained language model. We present the methods for gathering data, aligning texts, and removing preprocessing noise and translation errors automatically. We also train machine translation models based on this dataset to assess the quality of the corpus. Our models perform comparably to Google Translation API (as of May 2020) for Thai–English and outperform Google when the Open Parallel Corpus (OPUS) is included in the training data for both Thai–English and English–Thai translation. The dataset is available for public use under CC-BY-SA 4.0 License. The pre-trained models and source code to reproduce our work are available under Apache-2.0 License.

✉ Attapol T. Rutherford
attapol.t@chula.ac.th

Lalita Lowphansirikul
lalital_pro@vistec.ac.th

Charin Polpanumas
charin.polpanumas@datatouille.org

Sarana Nutanong
snutanon@vistec.ac.th

¹ School of Information Science and Technology, Vidyasirimedhi Institution of Science and Technology, Rayong, Thailand

² PyThaiNLP, Bangkok, Thailand

³ Department of Linguistics, Chulalongkorn University, Bangkok, Thailand

⁴ Teaching and Learning Thai as a Foreign Language Group, Bangkok, Thailand

Keywords Machine translation · Parallel corpus · Pretraining · Transformer · Thai language

1 Introduction

Machine translation (MT) techniques have advanced rapidly in the last decade with many practical applications, especially for high-resource language pairs, for instance, English–German, English–French (Ott et al. 2018) and Chinese–English (Hassan et al. 2018). All of the modern machine translation models require a large number of parallel segments to train and benchmark on. Examples of these parallel datasets include News Commentary Parallel Corpus,¹ UN Parallel Corpus (Ziems et al. 2016), Europarl (Koehn 2005) and the ParaCrawl corpus (Esplà et al. 2019). However, English–Thai is a low-resource language pair. Insufficient number of training examples is found to directly deteriorate translation quality (Koehn and Knowles 2017) as current state-of-the-art models (Bahdanau et al. 2014; Gehring et al. 2017; Vaswani et al. 2017) require a substantial amount of training data to perform well. Therefore, we curate this dataset of approximately one million English–Thai sentence pairs to solve the challenges of both quantity and diversity of English–Thai machine translation data.

Constructing an English–Thai machine translation dataset entails several difficulties: the costs of acquiring high-quality translated segment pairs, the complexity of segment alignment due to the ambiguity of Thai sentence boundaries, and a limited number of web pages and documents with English–Thai bilingual content. Currently, the largest source of English–Thai segment pairs is the Open Parallel Corpus (OPUS) (Tiedemann 2012). It comprises parallel segments for many language pairs including English–Thai. However, the contexts of those segment pairs are limited to subtitles (OpenSubtitles (Lison and Tiedemann 2016), QED (Abdelali et al. 2014)), religious texts (Bible (Christodouloupoulos and Steedman 2015), JW300 (Agić and Vulić 2019), Tanzil²), and open-source software documentation (Ubuntu,³ KDE4,⁴ GNOME⁵).

In order to build an English–Thai machine translation dataset with a sufficient number of training examples from a variety of domains, we curate a total of 1,001,752 segment pairs from web-crawled data, government documents, texts generated by a pretrained language model, and publicly available datasets for NLP tasks in English. For some data sources, we used embedding-based sentence aligner to align potentially bilingual sentences. For the other sources, we hire professional translators only for difficult text and crowdsource translation for easy text to save on translation cost similar to Zaidan (2012). Using OPUS and our dataset, we train machine translation models based on transformer (Vaswani et al. 2017) and

¹ <http://www.casmacat.eu/corpus/news-commentary.html>.

² <http://opus.nlpl.eu/Tanzil.php>.

³ <http://opus.nlpl.eu/Ubuntu.php>.

⁴ <http://opus.nlpl.eu/KDE4.php>.

⁵ <http://opus.nlpl.eu/GNOME.php>.

compare the model performance with Google and AI-for-Thai translation services. We use Thai–English IWSLT 2015 (Cettolo et al. 2015) as a benchmark dataset and BLEU (Papineni et al. 2002) and chrF3 (Popović 2015) as the evaluation metrics. BLEU is widely used to evaluate translation quality by comparing translated segments with ground-truth segments. chrF3 is suitable for evaluating Thai translation because the Thai language does not mark word boundaries. Therefore, chrF3 calculation does not rely on a particular choice of word segmentation technique. Higher BLEU and chrF3 scores indicate better correspondence between the results and ground-truth translation. Our models are comparable to Google Translation API (as of May 2020) for Thai → English and outperform for both directions when OPUS is included in the training data.

Our English–Thai machine translation dataset⁶ and pre-trained machine translation models⁷ are publicly available on our GitHub repositories. We also present additional datasets for other Thai NLP tasks such as review classification and sentence segmentation, which are created as a result of building the machine translation dataset.

The rest of the paper is organized as follows. In Sect. 2, we first describe the sources from which segment pairs are retrieved for our dataset. After that, we detail the methods to obtain segment pairs, verify translation quality, and filter out noisy segment pairs. In Sect. 3, we exhibit the statistics of our resulting dataset, namely number of segments, number of tokens, and the distribution of segment pair similarity scores. Sect. 4 presents the results of our experiments where we train machine translation models on OPUS and our dataset and evaluate the performance on IWSLT 2015, OPUS, and our dataset. In Sect. 5, we discuss the challenges in building the English–Thai machine translation dataset and explore the opportunities to further improve the methodology to obtain a larger and better dataset. Our work is then concluded in Sect. 6.

2 Data sources

We collect and generate over one million English–Thai segment pairs from five data sources and preprocess them for English–Thai and Thai–English machine translation tasks. The resulting dataset statistics after a pipeline of preprocessing and filtering are summarized in Table 4.

2.1 Publicly available datasets

We use English segments from following public datasets for natural language processing (NLP) and natural language understanding (NLU) tasks as source segments. These datasets are translated to Thai by professional and crowdsourced translators.

⁶ https://github.com/vistec-AI/dataset-releases/releases/tag/scb-mt-en-th-2020_v1.0.

⁷ https://github.com/vistec-AI/model-releases/releases/tag/SCB_1M+TBASE_v1.0.

- Taskmaster-1 (Byrne et al. 2019) is a dataset of 13,215 task-based dialogs in 6 domains: ordering pizza, making auto repair appointments, scheduling rides, ordering movie tickets, ordering coffee drinks and making restaurant reservations. The dialogs are created in both written and spoken English.
- The National University of Singapore (NUS) SMS Corpus (Chen and Kan 2011) is a collection of 67,093 SMS messages written by Singaporeans, mostly NUS students. The style of writing is informal and contains so-called Singlish dialect of English.
- Mozilla Common Voice⁸ is a crowdsourced collection of 61,584 voice recordings in various languages. We use the English transcriptions as the source segments. The dataset has segments of both written and spoken English.
- Microsoft Research Paraphrase Identification Corpus (Dolan and Brockett 2005) contains 5801 English segment pairs from news sources. Each segment pair has a binary label of whether they paraphrase each other (that is, are semantically equivalent) or not.

2.2 Generated product reviews

We generate 372,534 product reviews in English using conditional transformer language model called CTRL (Keskar et al. 2019) and use them as the source segments. We choose to generate English data instead of Thai data because the cost for professionally translating English to Thai is lower. The conditional transformer language model was trained on multiple domains such as Amazon reviews, Wikipedia, Project Gutenberg and Reddit. CTRL can generate texts with content and style specified by the control codes. For our dataset, we specified the following conditions:

- The content generated must be in the product review domain.
- The generated reviews must represent sentiments ranging from mostly dissatisfied to mostly satisfied (1–5 scale).
- The length of each generated review is limited to less than 150 tokens. Incomplete segments as a result of the generation process are filtered out.

The median number of English segments in a review is 4 segments. The maximum number of segments per review is 19 segments

2.3 Wikipedia

Wikipedia consists of articles about various topics such as biographies, events, organizations and places. Articles are written and edited by crowdsourced contributors. We obtain 6,047,512 articles in English Wikipedia and 136,452 articles in Thai Wikipedia. We hypothesize that there are a number of articles among them that can be treated as parallel documents.

⁸ <https://voice.mozilla.org/en>.

2.4 Web crawling

Large machine translation datasets such as Paracrawl (Esplà et al. 2019) are created from scraping websites with parallel texts. We gather domains of possible parallel websites from three sources:

- Paracrawl: We aggregate the TMX files from 23 language pairs. The total number of domains listed is 208,349. The total number of URLs is approximately 12.8 million URLs. We directly substitute ISO 639-1, 639-2T, 639-2B language codes appeared in the URLs of non-English language code (e.g. /de/, /ger/, /es/, /spa/) to Thai language code (e.g. /th/, /tha/), and send HTTP request to verify whether the HTTP request of modified URL with Thai language code response with HTTP status 200. Out of 208,349 domains from 23 language pairs of Paracrawl, we found that 1047 domains have both English and Thai content.
- Top 500 Thai Websites according to Alexa.com⁹: We hypothesize that websites with high traffic volume are more likely to have pages both in Thai and English.
- Other specific bilingual websites such as Asia Pacific Defense Forum, Ministry of Foreign Affairs, and websites of various embassies in Thailand that provide sizable amounts of English–Thai content.

The data obtained by this method will be packaged under CC-BY-SA 4.0 License, but we do not own any of the text that has been extracted. We will comply to legitimate requests by removing the affected sources from the next release of the corpus.

2.5 Thai government documents

Official government documents in Thai and English in PDF format are obtained from their respective organizations:

- The Constitution of the Kingdom of Thailand 2017 (B.E. 2560)
- The Thailand Penal Code
- The Thailand Civil and Commercial Code
- Thailand’s Labour Relations Act 1975 (B.E. 2518)
- Thailand’s First through Twelfth National Economic and Social Development Plans (B.E.2504 - 2564; 1961–2021)
- Economic Outlook and Performance Report
- Social Outlook Report
- Gross Domestic Product report
- National Income of Thailand report
- Oil plan 2015–2036 (B.E. 2558 - 2579)
- Thailand 20-Year Energy Efficiency Development Plan 2011–2030 (B.E. 2554 - 2573)
- Alternative Energy Development Plan 2015–2036 (B.E. 2558 - 2579)

⁹ <https://www.alexa.com/topsites/countries/TH>.

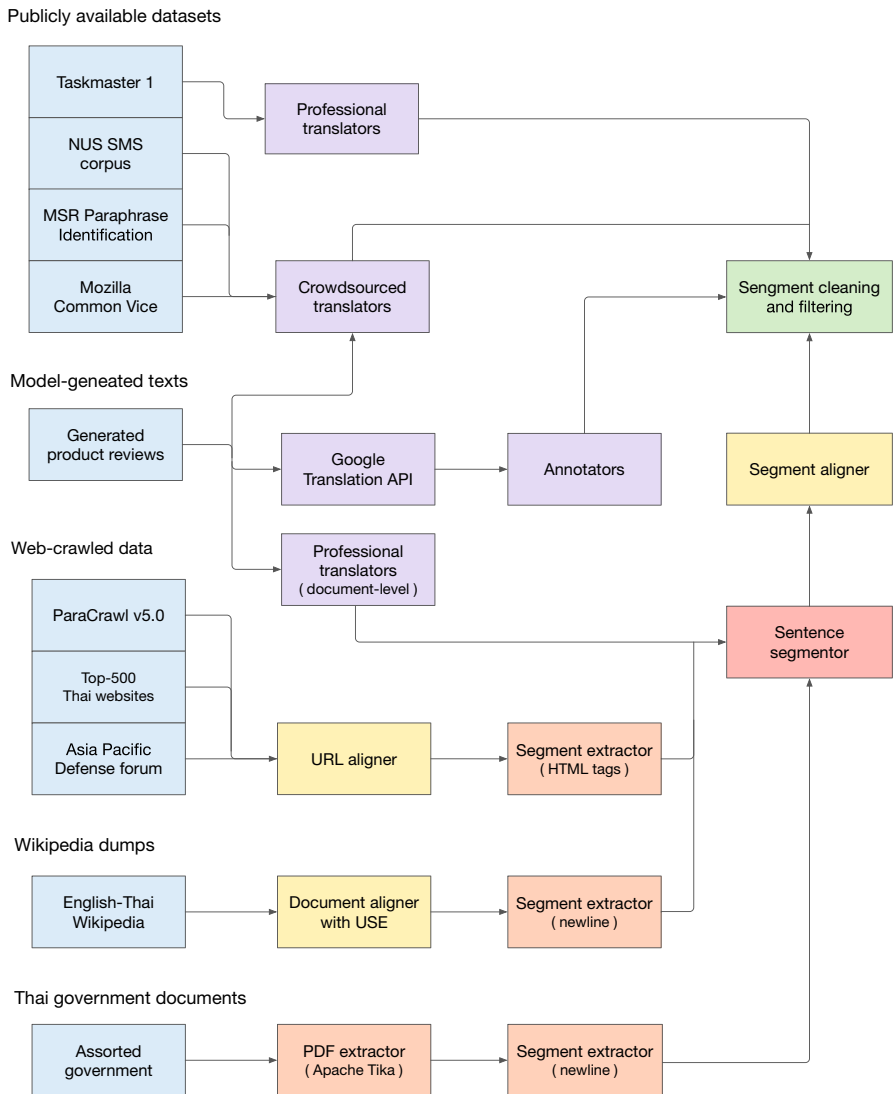


Fig. 1 Preprocessing flow for each data source. The color of boxes in light blue indicates data sources, purple indicates target language translation acquisition method, yellow indicates document/segment aligner, orange indicates text segment extractor, red indicates sentence segmentor, and green indicates segment pairs filter/cleaner

- Thailand Power Development Plan 2015–2036 (B.E. 2558 - 2579)
- Sustainable Future City Initiative Guideline for SFCI Cities

3 Data preparation methodology

The data sources come in many formats and require extensive preprocessing until they are filtered and converted to parallel sentences that are ready for machine translation models. It is our goal to extract text segments, break them into sentences, and align them with their translation. Some data sources require more preprocessing than others. Some require manual translation, but some only require aligning Thai and English sentences. This section describes the details of each preprocessing step, which is also summarized in Fig. 1.

3.1 Translation of English segments

We only translate English segments into Thai and not the other way around because the cost for translation is lower than the other direction. One way to create segment pairs is to employ various translation methods. We employ three approaches to get the translation: professional translation, crowdsourced translation and Google Translation API.

For professional translation, we employ 25 professional translators to translate 13,215 conversations of the Taskmaster-1 dataset at the rate of 1 Thai Baht (THB) per utterance and 43,374 generated product reviews at the rate of 0.3 THB per sentence. Secondly, we use a crowdsourcing platform to distribute English-to-Thai translation tasks for NUS SMS, Mozilla Common Voice, and Microsoft Research Paraphrase Identification, and 21,590 generated product reviews. The translation fee is set relatively low because the language in the corpora does not require technical knowledge or complicated translation. Plus, the translation from English to Thai is cheaper than the other way around. The total cost of translation still comes out quite high at around 30,000 USD. The translation project (including the translation quality management and the translation process itself) lasts around 8 months.

Manual translation is costly and time-consuming; therefore, we also opt for human-in-the-loop translation to make the total cost more feasible. We first use Google Translation API to translate 307,570 generated English product reviews to Thai and employ professional annotators to assess the quality of each product review. We ask the annotators to classify whether the product reviews translation should be accepted or rejected. The criteria are fluency and adequacy of the translation. One product review may have several segments, but we only include segments from product reviews that are labeled as acceptable. The annotation fee is set at 1.5 THB per review.

3.2 Alignment of existing English-Thai segments

Some of our data sources already have parallel English-Thai segments, but they are not aligned sentence by sentence, which is the form required by most machine translation models. So we must extract the segments, break them into sentences, and then align them into parallel sentence pairs.

3.2.1 Text segment extraction

3.2.1.1 Paracrawl Corpus release v5.0 (September 2019) We use the web crawling module from bitextor (Esplà and Transducens 2009) to crawl the websites and perform language detection to filtered out the pages whose contents are in neither English nor Thai. To avoid being shut off due to excessive queries to the same domain, we submit HTTP HEAD request with 5 target URLs for each target non-English language (i.e. 25 synthesized URLs in total for each domain name). Then, for those domains that the synthesized URLs return STATUS 200, we perform HTTP GET request to detect pages written in Thai.

After we obtained a list of domains that contains Thai content, we start crawling from those domains with the crawler from bitextor (wget-crawler). We then perform document alignment on crawled data of each domain name based on edit distance of tokens in URLs. A token in this case is defined by a group of characters separated by/except for the protocols (http:, https: and so on). Two URLs with edit distance equal to one token are paired up; for example, two URLs that differ only in the language code tokens. We successfully aligned 23,528 document pairs.

3.2.1.2 Top-500 Thai websites . We crawl all bilingual web pages linked in the sitemaps of the top 500 websites in Thailand, based on the ranking website Alexa.com. Similar to above, if a URL contains an English or Thai language code, we substitute the language code with /en/ or /th/ and verify if the document pair contains content both in English and Thai. The total number of aligned documents crawled is 246,868 page pairs.

3.2.1.3 Wikipedia To create parallel documents from Wikipedia pages, we align English and Thai articles based on their titles by transforming them into dense vectors using multilingual universal sentence encoder (Yang et al. 2019) and find cosine similarity. Out of all English and Thai articles, we find 13,853 articles that we consider parallel documents.

3.2.1.4 Government documents in PDF format We extract segments from aligned government documents in PDF format with Apache Tika.¹⁰ Character errors in extracted Thai texts are fixed with handcrafted rules.¹¹

3.2.1.5 Thai translation of generated product reviews Professionals translate 43,374 generated product reviews to Thai. Since the translation is document-level, we need to extract segments from the source reviews and translated reviews in order to obtain the alignment at segment-level.

¹⁰ <https://tika.apache.org/>.

¹¹ See <https://github.com/vistec-AI/pdf2parallel>.

Table 1 The precision, recall and F1 score for sentence boundary token of CRF-based sentence segmentor models trained and validated on different datasets

| Training set | Validation set | Sentence boundary token | | | Accuracy |
|-------------------------------|----------------|-------------------------|--------|----------|----------|
| | | Precision | Recall | F1 score | |
| TED + Orchid + Product review | TED | 0.66 | 0.77 | 0.71 | 0.78 |
| TED + Orchid + Product review | Orchid | 0.73 | 0.66 | 0.69 | 0.82 |
| TED + Orchid + Product review | Product review | 0.98 | 0.95 | 0.96 | 0.96 |

3.2.2 Sentence breaking

It is quite straightforward to break an English text segment into English sentences. Most English sentence boundaries are marked by punctuations although some ambiguous cases exist. We use NLTK (Loper and Bird 2002) for English sentence breaking, which utilize an extensive set of regular expressions and rules.

Thai sentence breaking, however, is more difficult because Thai text does not have word boundaries or sentence boundaries that are clearly marked by punctuation. Spaces in Thai are added to separate certain constituents such as dependent clauses, coordinated nouns, and sentences. We take advantage of this linguistic fact to formulate sentence breaking problem as a space disambiguation/classification task. Each space in the sequence is tagged as positive (sentence boundary) or negative (not a sentence boundary). Previous methods for breaking Thai sentences include using Winnow and MaxEnt models (Slayden et al. 2010; Tangsirirat et al. 2013) with character-based context features. These methods do not take advantage of sequential information in the sequence of spaces. Here, we propose a CRF-based sentence-breaking model that uses lexical features interspersed within a sequence of spaces.

The labeled data come from a few different sources. Like the previous studies, we use ORCHID corpus as the primary gold standard training data (Sornlertlamvanich et al. 1997). The corpus contains 23,125 text segments which are also marked with sentence boundaries. We create additional training data without manual sentence boundary annotation through artificially generated parallel sentences. From the Generated Product reviews, we use a total of 217,482 segments that are translated by Google Translate API and verified by humans as additional data since we know the sentence boundaries marked in the English texts. We add sentence boundaries between the automatically translated and manually verified Thai translation sentences, which are stitched back together to form a long string as training data. Moreover, we also use a portion of TED Transcript data that have Thai translation. We extract 136,463 parallel utterances. We treat each utterance as a sentence. The dataset consists of 3,258,276 words or 12,789,186 characters. In terms of label distribution, the dataset comprise 3,121,629 spaces that are not sentence boundaries (negative label) and 136,647 spaces that are sentence boundaries (positive label). We shuffle the data and assign 80% to be the training set and the other 20% to be the test set.

We train a Conditional Random Fields (CRF) model to tag a sequence of space tokens as sentence boundary tokens or not. We tokenize texts into Thai words using the dictionary-based maximal matching tokenizer (so-called *newmm* engine) of PyThaiNLP (Phatthiyaphaibun et al. 2020) and generate *template features* for CRF. In particular, we extract unigram, bigram and trigram features within a sliding window of two timesteps (before and after the space token to predict if it is a sentence boundary or not. To illustrate, the bigram features are $w_{t-3:t-2}$, $w_{t-2:t-1}$, $w_{t-1:t+1}$, $w_{t+1:t+2}$, $w_{t+2:t+3}$. In addition, we also featurize words that are often found to be sentence starters or sentence enders and apply the same feature extraction. In particular, we use ending particles, discourse connectives that take ending positions, and demonstrative pronouns as features.

Our CRF-based sentence breaker achieves F1 score of 0.71 on TED dataset, 0.69 on ORCHID dataset, and 0.96 on product review dataset. Our model performance is summarized in Table 1. The training code is available at <https://github.com/vistec-AI/crfcut>. The TED dataset is the most difficult because transcription of spoken language, does not always conform with certain writing standards found in written text such as ORCHID and Product Review corpora.

3.2.3 Sentence alignment

For each pair of aligned documents, we have two approaches in aligning segments and sentences. The first approach is applicable for documents crawled from the web. We segment the content in the documents by HTML tags (e.g. `<p >`, ``, and `<h1 >`). `<div >` is not considered by this process because the textual portion of the block is usually marked by `<p >`. In this case, we extract only the child tags are for textual content, and discard all irrelevant tags. As a result, the extracted segments of a HTML page will be a flattened list of tags with textual portion inside. For instance, the code block

```
<div>
  <h1>Header</h1>
  <a href="#">link</a>
  <div>
    <p>paragraph</p>
  </div>
</div>
```

will be converted to [`<h1 > Header </h1 >`, `<p > paragraph </p >`].

All content within a tag is treated as one segment. We then choose only document pairs that have the same number of equivalent tags and align the segments in order. The downside of this approach is that we might end up with multiple segments per tag.

For the segment alignment step, we use the HTML tags to guide the alignment. Given a pair of flattened lists of tags from aligned webpage, we will reject the pair if it doesn't have the same number of equivalent tag names in the same order. The difference from Bitextor (Esp  -Gomis and Forcada 2009) is that our approach considers tags in a flat structure, not a nested structure. In addition, we will use

semantic criteria afforded by sentence embeddings to perform the alignment as will be explained in the next section.

The second approach is to use sentence segmenter described in the previous section to segment Thai texts and NLTK sentence segmenter (Loper and Bird 2002) to segment English texts then align them based on semantic similarity. We considered the Gale–Church algorithm, which assumes that aligned sentences should have correlated sentence lengths and that sentences are translated in order or not translated at all (Gale and Church 1993). We found that the web crawl data contain many aligned sentences that do not follow the same ordering as the source sentences, so we require an aligner that operates on the semantics of the sentences. Also, we experimented with lexicon-based sentence aligners (e.g. Varga et al. (2007) or Ma (2006)) and found too many misaligned sentences upon inspection. We found that the translation from web crawl is either noisy or performed beyond the lexical level, which is assumed by these lexicon-based sentence aligners. And these methods were never tested on the Thai language, and the alignment quality might be sensitive to specific language pairs. Therefore, we employ an embedding-based method similar to Thompson and Koehn (2019), which vectorizes sentences and allows many-to-one alignment. We align each English sentence with a concatenation of one to three contiguous Thai sentences, and each sentence is allowed to be aligned in any ordering. We use multilingual Universal Sentence Encoder (Yang et al. 2019) trained on 13 languages, including English and Thai, to transform each sentence into a 512-dimension dense vector. Then we compute cosine similarity of all pairs of English and Thai concatenated sentences. For each English sentence, we select the sentence or the group of sentences that receives the highest cosine similarity score. But the similarity score must also exceed the threshold. We use a different cosine similarity threshold for segments from each domain. For example, texts retrieved from web crawling have a relatively higher threshold of 0.7 as we see a higher rate of misalignment, whereas the segment pairs from Thai government documents have the threshold of 0.5 as they follow set patterns and are easier to align.

3.3 Cleaning and filtering

We clean text by performing normalizing NFKC Unicode text, replacing HTML entity and number code (e.g. " and ") with corresponding ASCII characters, removing redundant spaces, and standardizing quote characters. Note that emojis and emoticons are not filtered out from the texts.

Since we obtain our sentence pairs by different sources and approaches with varying degrees of quality, we have to filter out some sentence pairs that are not parallel to each other. The Paracrawl project has its own sentence-pair cleaner that we could repurpose for this project. However, it requires a lot of existing high-quality parallel corpus to train the cleaner. Instead, we utilize the recent advancement in multilingual pre-trained language models. We filter sentence pairs by a set of handcrafted rules and, more importantly, text similarity based on

Table 2 The thresholds of parameters we used in filtering segment pairs for each sub-dataset

| Sub-dataset | Threshold of word tokens | | Min character percentage | | Threshold of th to en tokens ratio | Minimum cosine similarity |
|------------------------------|--------------------------|----------|--------------------------|------|------------------------------------|---------------------------|
| | th | en | th | en | | |
| task_master_1 | [3, 500] | [3, 500] | 0.50 | 0.50 | [0.1, 2.0] | 0.20 |
| generated_reviews_translator | [2, 500] | [2, 500] | 0.40 | 0.40 | – | 0.50 |
| nus_sms | [1, 500] | [1, 500] | – | – | [0.06, 6.0] | 0.10 |
| msr_paraphrase | [3, 500] | [3, 500] | 0.65 | 0.10 | [0.30, 2.0] | – |
| mozilla_common_voice | [2, 500] | [1, 500] | 0.55 | 0.50 | [0.13, 11.0] | 0.30 |
| generated_reviews_crowd | [2, 150] | [1, 500] | 0.50 | 0.50 | – | 0.35 |
| generated_reviews_yn | [2,500] | [4, 500] | 0.50 | 0.50 | – | 0.40 |
| assorted_government | [4, 500] | [4,500] | 0.50 | 0.25 | (0,4.0) | 0.30 |
| thai_websites | [3, 500] | [1, 500] | 0.55 | 0.45 | (0, 8.5] | 0.10 |
| paracrawl | [5, 500] | [3, 500] | 0.50 | 0.50 | [0.05, 2.3] | 0.50 |
| wikipedia | [5, 500] | [5, 500] | 0.50 | 0.50 | [0.5, 1.45] | 0.70 |
| apdf | [6,500] | [5, 500] | 0.50 | 0.50 | – | 0.40 |

multilingual Universal Sentence Encoder.¹² This approach has been found to work better than Paracrawl’s Bicleaner for low-resource languages (Chaudhary et al. 2019). For each dataset, we define a set of thresholds for the following handcrafted rules to filter out low-quality segment pairs:

- Percentage of English or Thai characters in each English or Thai segment; for instance, Thai segments with lower percentage of Thai characters are most likely not actually Thai segments but segments from other languages that have been mistakenly crawled
- Minimum and maximum number of word tokens for Thai and English segment. We use *newmm* tokenizer from PyThaiNLP (Phatthiyaphaibun et al. 2020) to tokenize Thai words, and NLTK (Loper and Bird 2002) to tokenize English words. Spaces are excluded from the token counts.
- Ratio of word tokens between English and Thai segments; for example, a pair of segment with 100 tokens for English and 5 tokens for Thai will be filtered out from the resulting dataset.

4 Resulting datasets

We collect segment pairs from 12 sources and perform the text processing procedures described in Methodology. After cleaning and filtering, we amass a total of 1,001,752 sentence pairs (Table 3). Around 35% of the corpus is obtained from

¹² The source code and thresholds used for the preprocessing can be found at: https://github.com/vistec-AI/thai2nmt_preprocess.

Table 3 Number of segment pairs categorized by data source and method to obtain parallel sentence pairs

| Method | Sub-dataset | Number of sentence pairs |
|--|-----------------------------|--------------------------|
| Professional translators | task_master_1 | 222,733 |
| | generated_review_translator | 133,330 |
| Crowd-sourced translators | nus_sms | 43,750 |
| | msr_paraphrase | 10,371 |
| | mozilla_common_voice | 33,797 |
| | generated_review_crowd | 24,587 |
| Annotation by Translators | generated_review_yn | 280,208 |
| Sentence alignment on PDF documents | assorted_government | 25,398 |
| Sentence alignment on web-crawled data | thai_websites | 120,280 |
| | paracrawl | 60,039 |
| | wikipedia | 33,756 |
| | apdf | 13,503 |
| | | 1,001,752 |

professional translators, which represent the true gold standard dataset. The “useable” sentence pairs obtained from web scraping account for 22% of our dataset. This proportion is far lower than expected because we apply stringent alignment and filtering procedures to ensure that the parallel sentences do not introduce noise in model training process.

As an additional measure for ensuring data quality, we examine the corpus statistics of each data source (Table 4). The mean and median number of tokens are similar for each sub-dataset. The vocabulary size of English data is larger than Thai data, possibly because English is morphologically richer than Thai.

In addition, we automatically check the translation quality by using Universal Sentence Encoder. We assume that good translations should be high in their cosine similarity in the USE space. All of our sub-datasets score higher than 0.7 on average except for *task_master_1* (self-conversation data) and *nus_sms* (mobile text-messaging data) (Fig. 3). This is understandable because these two datasets contain colloquial language, which is not what USE is trained on.

5 Experiments

To test whether our newly created corpus can be used to train machine translation models effectively, we compare the performance of the model trained on this corpus alone and the performance of Google Translation and AI-for-Thai, which are strong baselines. Further, we compare the performance of the model trained on this corpus alone versus one trained with additional parallel sentences from OPUS.

Table 4 Number of segment pairs, Thai/English word tokens, unique word tokens and distribution of English and Thai word tokens in segments for each sub-dataset

| Sub-dataset name | | Tokens | Vocab size | Token distribution | | |
|------------------------------|----|------------|------------|--------------------|--------|------------|
| | | | | Mean | Median | (min, max) |
| task_master_1 | en | 2,615,760 | 32,888 | 11.74 | 10 | (1, 211) |
| | th | 2,349,135 | 20,406 | 10.55 | 8 | (3, 203) |
| generated_reviews_translator | en | 2,128,286 | 32,025 | 15.96 | 14 | (1, 102) |
| | th | 1,974,424 | 22,109 | 14.81 | 13 | (2, 117) |
| nus_sms | en | 538,584 | 33,816 | 12.31 | 10 | (1, 171) |
| | th | 561,907 | 13,329 | 12.84 | 10 | (1, 172) |
| msr_paraphrase | en | 231,897 | 18,191 | 22.36 | 22 | (3, 46) |
| | th | 219,682 | 15,776 | 21.18 | 21 | (3, 52) |
| mozilla_common_voice | en | 325,856 | 17,377 | 9.64 | 9 | (2, 28) |
| | th | 288,066 | 15,578 | 8.52 | 8 | (1, 54) |
| generated_reviews_crowd | en | 441,804 | 13,246 | 17.97 | 16 | (3, 89) |
| | th | 391,505 | 12,169 | 15.92 | 14 | (2, 91) |
| generated_reviews_yn | en | 4,429,469 | 37,202 | 15.81 | 14 | (2, 104) |
| | th | 3,909,029 | 26,261 | 13.95 | 12 | (3, 96) |
| assorted_government | en | 1,711,174 | 25,139 | 67.37 | 63 | (5, 500) |
| | th | 1,931,200 | 25,802 | 76.04 | 64 | (4, 441) |
| thai_websites | en | 9,934,983 | 117,267 | 82.60 | 70 | (3, 543) |
| | th | 11,105,989 | 85,096 | 92.33 | 80 | (1, 455) |
| wikipedia | en | 1,655,315 | 54,173 | 49.04 | 47 | (6, 226) |
| | th | 1,839,488 | 40,570 | 54.49 | 40 | (5, 272) |
| paracrawl | en | 1,688,408 | 56,196 | 28.12 | 19.0 | (5, 316) |
| | th | 1,691,030 | 39,035 | 28.17 | 19.0 | (3, 322) |
| apdf | en | 685,864 | 25,516 | 50.79 | 46 | (6, 303) |
| | th | 736,931 | 15,301 | 54.58 | 49 | (5, 331) |

5.1 Data

For the experiments, we use the preprocessed and filtered segment pairs summing up to 1,001,752 pairs, described in detail in the previous section. We set the ratio for training/validation/test sets to 80/10/10. The dataset is split and sampled in a stratified manner with respect to their sources, so every sub-dataset is represented in all three splits. We also ensure that no exact duplicate sentences exist within the same language shared in validation and test sets to prevent data leakage.

As additional training data, we use approximately five million parallel English–Thai sentences from OPUS (Tiedemann 2012), an open source parallel corpus. Out of 9 English–Thai parallel datasets currently listed in OPUS, we use the following 6

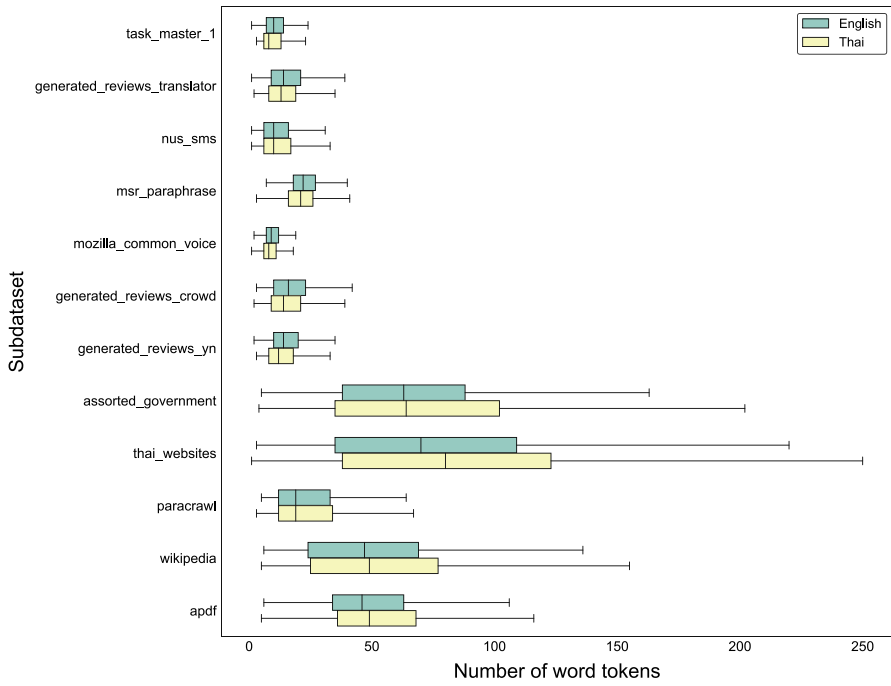


Fig. 2 The distributions of segment lengths for each sub-dataset. The data extracted from websites tends to be longer as they usually span longer than one sentence

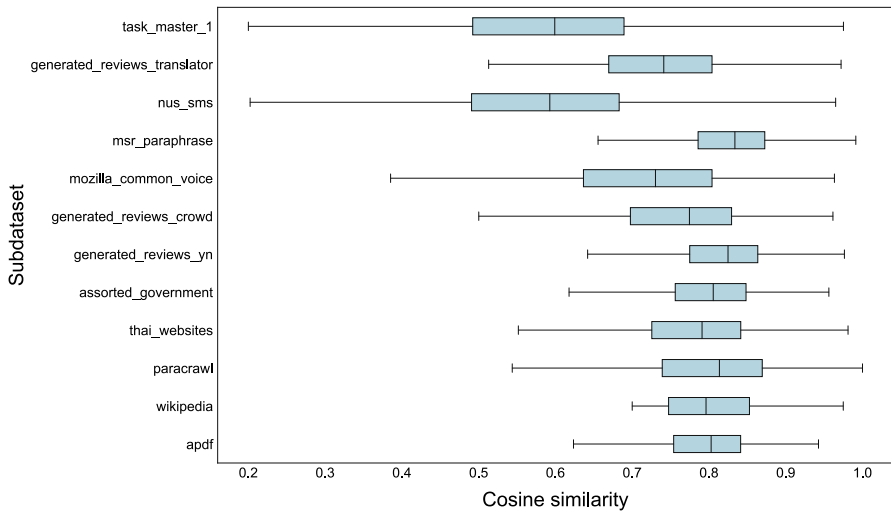


Fig. 3 The distributions of cosine similarity scores between Thai segments and their corresponding English segments for each sub-dataset

datasets: OpenSubtitles (Lison and Tiedemann 2016), Tatoeba,¹³ Tanzil,¹⁴ QED (Abdelali et al. 2014), Ubuntu and GNOME. The total number of segment pairs is 3,715,179. Then, we perform text cleaning as discussed the previous section and additionally set Thai/English character ratio limit up to 0.1 and number tokens up to 500 for each segment. We also remove exact duplicate pairs. The resulting datasets contain 3,318,153 sentence pairs in total. The ratio for training/validation/test sets is 80/10/10.

As a held-out test set, we use Thai–English IWSLT 2015 evaluation dataset (Cettolo et al. 2015), which contains parallel transcriptions of TED talks where the source language is Thai and target language is English. The number of sentences pairs is 4242 from 46 parallel TED talk transcriptions. We use IWSLT 2015 test sets from 2010 to 2013. We manually tokenize the Thai version of the data based on BEST 2010 guidelines, which are used to train most Thai tokenizers (Kosawat et al. 2009).

5.2 Models

We use the transformer (Vaswani et al. 2017), a supervised neural machine translation model, implemented in the Fairseq toolkit (Ott et al. 2019) as our NMT model in both English-to-Thai and Thai-to-English direction. We train transformer models with 6 encoder blocks, 6 decoder blocks, 512-dimensional embeddings, and 2,048 feed forward hidden units. The dropout rate is set to 0.3. The embeddings of decoder input and output are shared. The maximum number of tokens per mini-batch is 9750. The optimizer is Adam with the initial learning rate of 10^{-7} and weight decay rate of 0.0. The learning rate has an inverse squared schedule with warmup for the first 4,000 updates. Label smoothing of 0.1 is applied during training. The criteria for selecting the best model checkpoint is label-smoothed cross-entropy loss.

We also explore different ways of tokenization in both translation directions. Thai texts are either tokenized with PyThaiNLP’s dictionary-based word-level tokenizer (the ‘newmm’ engine) or SentencePiece tokenizer (Kudo and Richardson 2018). English texts are either tokenized with word-level Moses tokenizer or SentencePiece tokenizer that is trained on the training set itself. We experiment with all four combinations of tokenizations and both translation directions (Thai → English, and English → Thai).

When training transformers, the maximum number of tokens for each batch is set to 9750. The number of epochs for transformer is set to 150. All the models in this experiment are trained on NVIDIA V100 GPU with mixed-precision training (fp16) and gradient accumulation for 16 steps.¹⁵ For model decoding, the model checkpoint selected is the epoch with minimum label-smoothed cross-entropy loss in the development set. The beam width is set to 4.

¹³ tatoeba.org.

¹⁴ tanzil.net.

¹⁵ The source code used for the experiments can be found at: <https://github.com/vistec-AI/thai2nmt>.

5.3 Evaluation methods

SacreBLEU (Post 2018) is used to evaluation translation quality for Thai-to-English translation. BLEU4 and chrF3 are used to evaluate translation quality for English-to-Thai translation. However, the detail for computing BLEU scores for Thai output is not straightforward because we do not have the gold standard word segmentation. Therefore, BLEU scores must be computed with respect to a specific version of a Thai tokenizer. We de-tokenize the output with the inverse of its tokenizer and then tokenize the de-tokenized output the same way as the gold standard translation output for the best results. The version strings used for computing BLEU score for case-sensitive and case-insensitive are *BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.2.10* and *BLEU+case.lc+numrefs.1+smooth.exp+tok.13a+version.1.2.12*, respectively.

6 Experimental results

We report the evaluation results on the test set of our dataset, denoted as SCB_1M, and parallel English–Thai segments from OPUS. The total number of segment pairs from SCB_1M and OPUS test set are 100,177 and 297,874 respectively. We train models on each train set and cross validate on the test sets from 2 sources. The results suggest that our corpus can be used to train neural machine translation. We achieve BLEU scores of 39–42 in various configurations and both translation directions (Table 5).

It is noteworthy that Thai SentencePiece representation suffers substantially from out-of-domain problem. Unlike language models in the BERT family, our SentencePiece tokenizer is trained on the training set alone and not on other unlabeled data. On the other hand, English SentencePiece tokenizers do not suffer from the same

Table 5 Results on SCB_1M and OPUS test set for th → en and en → th of the transformer base models trained on either SCB_1M or OPUS train set

| Language pair | Token type | BLEU / chrF3 score (train set → test set) | | | |
|---------------|-------------|---|----------------------|----------------|----------------------|
| | | SCB_1M → SCB_1M | SCB_1M → OPUS | OPUS → OPUS | OPUS → SCB_1M |
| th → en | Word → word | 39.42 / 58.27 | 13.54 / 34.50 | 25.17 / 43.60 | 9.64 / 31.28 |
| | Word → sp | 38.41 / 57.57 | 13.96 / 34.75 | 25.58 / 44.07 | 10.50 / 33.48 |
| | sp → word | 39.09 / 58.02 | 6.87 / 23.85 | 26.09 / 44.48 | 5.80 / 22.15 |
| | sp → sp | 39.59 / 58.86 | 6.74 / 24.68 | 26.28 / 44.85 | 6.08 / 25.25 |
| en → th | word → word | 40.30 / 59.61 | 13.29 / 35.80 | 21.27 / 42.18 | 9.61 / 27.24 |
| | word → sp | 42.58 / 59.67 | 13.13 / 35.76 | 20.71 / 41.20 | 7.76 / 24.84 |
| | sp → word | 41.21 / 60.51 | 10.65 / 33.40 | 21.74 / 42.51 | 8.04 / 24.12 |
| | sp → sp | 42.94 / 59.91 | 11.33 / 33.19 | 20.53 / 40.65 | 5.43 / 20.13 |

sp denotes SentencePiece tokenization. Bold-faced numbers indicate statistical significance within the same cell (pairwise *t*-test with Bonferroni correction, $p < 0.05$)

Table 6 Evaluation results for Thai–English IWSLT 2015 test sets (tst2010-2013) for th → en and en → th of the transformer base model trained on SCB_1M, OPUS, and both

| Language pair | Token type | BLEU / chrF3 score on IWSLT 2015 when trained on . | | | |
|---------------|-------------|--|---------------|---------------|----------------------|
| | | SCB_1M | OPUS | OPUS_1M | SCB_1M + OPUS |
| th → en | Word → word | 14.32 / 39.55 | 20.81 / 45.46 | 16.36 / 39.51 | 25.55 / 48.63 |
| en → th | Word → word | 12.69 / 40.41 | 16.55 / 42.08 | 13.44 / 37.62 | 18.42 / 43.82 |

Bold-faced numbers indicate statistical significance within the same rows (pairwise *t*-test with Bonferroni correction, $p < 0.05$)

problem because English data only contain clear word and sentence boundaries, which are taken into account when training the SentencePiece tokenizer.

For SCB_1M test set, models trained on SCB_1M training set have consistently 4–8 times higher BLEU score than those trained on MT_OPUS. In similar manner, for MT_OPUS test set, models trained on MT_OPUS have 2–4 times higher BLEU score than those trained on SCB_1M. This suggests that diversity of domains in the training set greatly impacts the performance of the models.

We observe an improvement in performance when we use our dataset in conjunction with OPUS data (Table 6). We compare the performance of our baseline models trained on SCB_1M, OPUS, and both. When controlled for the amount of 1M sentences (SCB_1M vs OPUS_1M), the system achieves comparable BLEU score on the IWSLT 2015 test set. The OPUS_1M achieves a higher BLEU score possibly due to the fact that IWSLT 2015 is a collection of TED Talk transcripts which are in the same domain as OpenSubtitles (Lison and Tiedemann 2016), which constitute the majority of the OPUS dataset. The model trained on both OPUS (Tiedemann 2012) and our dataset achieves a 4-point increase in SacreBLEU for Thai-to-English translation and 2-point increase for English-to-Thai translation. This result suggests that our newly-created parallel corpus is effective for machine translation.

The baseline neural models trained on our datasets outperform other translation APIs (Table 7). We submitted the pre-processed data to Google Translation API (Neural Translation Model Predictions In Translation V3) on May 12, 2020 to obtain translations. Additionally, we submitted English sentences to the Translation API provided by AI-for-Thai,¹⁶ a new machine translation service in Thailand, to obtain translation in Thai on May 16, 2020. We evaluate only in English → Thai direction as at the moment AI-for-Thai provides only English → Thai translation. We report detokenized SacreBLEU (case-sensitive) for Thai → English direction, and BLEU4 (case-sensitive) for English → Thai direction. To assess the effectiveness of our corpus in training modern Machine Translation models, we conduct experiments on English → Thai and Thai → English machine translation systems trained on our dataset and the Open Parallel Corpus (OPUS) with different types of source and target token (i.e. word-level and subword-level). The evaluation results on Thai–English IWSLT 2015 test sets show that performance of our baseline models is on par with Google Translation API for Thai → English and

¹⁶ <https://www.aiforthai.in.th>.

Table 7 Results on Thai–English test sets (tst2010–2013)

| Language pair | Type | BLEU / chrF3 score for each system | | | | |
|---------------|---------|------------------------------------|--------------|---------------|---------------|----------------------|
| | | Google | AI-for-Thai | SCB_1M | OPUS | SCB_1M + MT OPUS |
| th → en | Cased | 14.19 / 43.60 | - | 17.14 / 42.73 | 27.94 / 50.63 | 28.39 / 51.10 |
| | uncased | 17.64 / 46.21 | - | 17.89 / 43.35 | 28.56 / 51.05 | 29.06 / 51.53 |
| en → th | Cased | 15.29 / 43.10 | 6.03 / 28.04 | 12.94 / 41.08 | 17.26 / 41.93 | 18.42 / 43.82 |

We submit detokenized source Thai segments to Google Translation API for translation into English. Our baseline model is transformer (base) where the source and target token is sub-word units computed by using the SentencePiece library. Bold-faced numbers indicate statistical significance within the same rows for the same metrics (pairwise *t*-test with Bonferroni correction, $p < 0.05$)

outperform for both direction when OPUS is included in the training data. Despite potential noise in our corpus, the model trained on this corpus manages to yield good translations comparable to a commercial system.

7 Discussion

In compiling this parallel corpus, we overcame a few challenges to ensure the quality. First, sentence breaking in Thai is not straightforward because Thai has no clear word or sentence boundary. We need to train a sentence breaker before moving on to translating Thai text or aligning sentences to their English translation.

Second, professional translation is prohibitively expensive, so we need to lower the cost by using crowdsourced translation. To the best of our knowledge, there exists no gold standard corpus created routinely by professional translators such as Europarl for Thai. We select the subset of the data that is easy enough for Thai native speakers with moderate English proficiency to translate. To control the quality of the translation, we filter out lower-quality translation by using text similarity threshold afforded by the Universal Sentence Encoder. Moreover, some crowdsourced translators might copy and paste source segments to a translation engine and take the results as answers to the platform. To further improve, we can apply techniques such as described in Zaidan (2012) to control the quality and avoid fraud on the platform.

Third, sentence alignment is complicated by noisy ML-based sentence breaking when dealing with data scraped from the web. Furthermore, even if sentence breaking is correct, one sentence in Thai may correspond to more than one sentence in English. In this work, we mitigate this problem by grouping Thai segments together before computing the text similarity scores. We then choose the combination with the highest text similarity score. This way, we can remedy the negative effects caused by wrong sentence breaking.

After taking these measures for controlling the quality of the data, we use the resulting parallel corpora to train a neural machine translation model. Owing to the

fact that we can train a model that achieves performance comparable to a commercial system, this is another piece of evidence that our parallel corpus meets the standard for machine translation.

8 Conclusion

We release an English–Thai parallel corpus comprising of over one million sentence pairs, including both written and spoken language. The corpus comprises text from various domains such as product reviews, laws, report, news, spoken dialogues, and SMS messages. We also release 4 additional datasets for Thai text classification tasks and Thai sentence segmentation task.

We present systems that deal with unsegmented Thai text, align Thai sentences with corresponding English sentences, and automatically filter out sentence pairs that do not pass the quality threshold. Translation evaluation is known to be subjective and an active area of research, but it is crucial for creating a parallel corpus in a semi-automatic manner, which we present here. In our future work, we will investigate different ways to evaluate sentence pairs obtained by scraping the web or by crowdsourcing translation.

To assess the effectiveness of our corpus in training modern Machine Translation models, we conduct experiments on English → Thai and Thai → English machine translation systems trained on our dataset and the Open Parallel Corpus (OPUS) with different types of source and target token (i.e. word-level and subword-level). The evaluation results on Thai–English IWSLT 2015 test sets show that performance of our baseline models is on par with Google Translation API for Thai → English and outperform for both direction when OPUS is included in the training data. Despite potential noise in our corpus, the model trained on this corpus manages to yield good translations comparable to a commercial system.

Acknowledgements This investigation is partially supported by the Digital Economy Promotion Agency Thailand under the infrastructure project code MP-62-003, Siam Commercial Bank, Special Task Force for Activating Research (STAR) Ratchadapiseksompoch Fund from Chulalongkorn university, and Research Grant for New Scholars from Thailand Research Fund (MRG6280175). We thank our data annotation partners Hope Data Annotations and Wang: Data Market; Office of the National Economic and Social Development Council (NESDC) through Phannisa Nirattiwongsakorn for providing government documents; Chonlapat Patanajirasit for training CRFCut sentence segmentation models on new datasets; Witchapong Daroontham for product review classification baselines; Pined Laohapiengsak for helping with sentence alignment using universal sentence encoder.

References

- Abdelali, A., Guzman, F., Sajjad, H., & Vogel, S. (2014). The AMARA corpus: Building parallel language resources for the educational domain. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), European Language Resources Association (ELRA), Reykjavik, Iceland, pp 1856–1862, http://www.lrec-conf.org/proceedings/lrec2014/pdf/877_Paper.pdf
- Agić Ž, V. I. (2019). JW300: A wide-coverage parallel corpus for low-resource languages. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics,

- Association for Computational Linguistics, Florence, Italy, pp 3204–3210. <https://doi.org/10.18653/v1/P19-1310>, <https://www.aclweb.org/anthology/P19-1310>
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. ArXiv 1409
- Byrne, B., Krishnamoorthi, K., Sankar, C., Neelakantan, A., Duckworth, D., Yavuz, S., Goodrich, B., Dubey, A., Cedilnik, A., & Kim, KY. (2019). Taskmaster-1: Toward a realistic and diverse dialog dataset. arXiv preprint [arXiv:190905358](https://arxiv.org/abs/190905358)
- Cettolo, M., Niehues, J., Stüker, S., Bentivogli, L., Cattoni, R., & Federico, M. (2015). The iwslt 2015 evaluation campaign
- Chaudhary, V., Tang, Y., Guzmán, F., Schwenk, H., & Koehn, P. (2019). Low-resource corpus filtering using multilingual sentence embeddings. In: Proceedings of the Fourth Conference on Machine Translation (Vol. 3: Shared Task Papers, Day 2), Association for Computational Linguistics, Florence, Italy, pp 261–266, <https://doi.org/10.18653/v1/W19-5435>, <https://www.aclweb.org/anthology/W19-5435>
- Chen, T., & Kan, M. Y. (2011). Creating a live, public short message service corpus: The nus sms corpus. *Language Resources and Evaluation*. <https://doi.org/10.1007/s10579-012-9197-9>.
- Christodouloupoulos, C., & Steedman, M. (2015). A massively parallel corpus: The bible in 100 languages. *Language Resources and Evaluation*, 49(2), 375–395. <https://doi.org/10.1007/s10579-014-9287-y>.
- Dolan, W. B., & Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In: Proceedings of the Third International Workshop on Paraphrasing (IWP2005), <https://www.aclweb.org/anthology/I05-5002>
- Esplà, M., & Transducens, G. (2009). Bitextor, a free/open-source software to harvest translation memories from multilingual websites
- Esplà, M., Forcada, M., Ramírez-Sánchez, G., & Hoang, H. (2019). ParaCrawl: Web-scale parallel corpora for the languages of the EU. In: Proceedings of Machine Translation Summit XVII Volume 2: Translator, Project and User Tracks, European Association for Machine Translation, Dublin, Ireland, pp 118–119, <https://www.aclweb.org/anthology/W19-6721>
- Esplà-Gomis, M., & Forcada, M. L. (2009). Bitextor, a free/open-source software to harvest translation memories from multilingual websites. Proceedings of MT Summit XII, Ottawa, Canada Association for Machine Translation in the Americas
- Gale, W. A., & Church, K. (1993). A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1), 75–102.
- Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. CoRR [arXiv:abs/1705.03122](https://arxiv.org/abs/1705.03122),
- Hassan, H., Aue, A., Chen, C., Chowdhary, V., Clark, J., Federmann, C., Huang, X., Junczys-Dowmunt, M., Lewis, W., Li, M., Liu, S., Liu, T. Y., Luo, R., Menezes, A., Qin, T., Seide, F., Tan, X., Tian, F., Wu, L., Wu, S., Xia, Y., Zhang, D., Zhang, Z., & Zhou, M. (2018). Achieving human parity on automatic chinese to english news translation. ArXiv abs/1803.05567
- Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., & Socher, R. (2019). Ctrl: A conditional transformer language model for controllable generation. [arXiv:1909.05858](https://arxiv.org/abs/1909.05858)
- Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. *MT Summit Citeseer*, 5, 79–86.
- Koehn, P., & Knowles, R. (2017). Six challenges for neural machine translation. In: Proceedings of the First Workshop on Neural Machine Translation, Association for Computational Linguistics, Vancouver, pp 28–39. <https://doi.org/10.18653/v1/W17-3204>, <https://www.aclweb.org/anthology/W17-3204>
- Kosawat, K., Boriboon, M., Choittrakool, P., Chotimongkol, A., Klaitthin, S., Kongyoung, S., Kriengkiet, K., Phaholphinyo, S., Purodakananda, S., Thanakulwarapas, T., & Wutiwiwatchai, C. (2009). Best 2009 : Thai word segmentation software contest. In: 2009 Eighth International Symposium on Natural Language Processing, pp 83–88. <https://doi.org/10.1109/SNLP.2009.5340941>
- Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Brussels, Belgium, pp 66–71, <https://doi.org/10.18653/v1/D18-2012>, <https://www.aclweb.org/anthology/D18-2012>

- Lison, P., & Tiedemann, J. (2016). OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), European Language Resources Association (ELRA), Portorož, Slovenia, pp 923–929, <https://www.aclweb.org/anthology/L16-1147>
- Loper, E., & Bird, S. (2002). Nltk: The natural language toolkit. In: Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics, Vol. 1, Association for Computational Linguistics, USA, ETMTNLP'02, p 63–70, <https://doi.org/10.3115/1118108.1118117>
- Ma, X. (2006). Champollion: A robust parallel text sentence aligner. In: LREC, pp 489–492
- Ott, M., Edunov, S., Grangier, D., & Auli, M. (2018). Scaling neural machine translation. ArXiv abs/1806.00187
- Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., & Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling. In: Proceedings of NAACL-HLT 2019: Demonstrations
- Papineni, K., Roukos, S., Ward, T., & Zhu, WJ. (2002). Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, pp 311–318, <https://doi.org/10.3115/1073083.1073135>, <https://www.aclweb.org/anthology/P02-1040>
- Phatthiyaphaibun, W., Chaovavanich, K., Polpanumas, C., Suriyawongkul, A., Lowphansirikul, L., & Chormai, P. (2020). Pythainlp/pythainlp: Pythainlp 2.1.4. <https://doi.org/10.5281/zenodo.3659277>.
- Popović, M. (2015). chrF: character n-gram F-score for automatic MT evaluation. In: Proceedings of the Tenth Workshop on Statistical Machine Translation, Association for Computational Linguistics, Lisbon, Portugal, pp 392–395, <https://doi.org/10.18653/v1/W15-3049>, <https://www.aclweb.org/anthology/W15-3049>
- Post, M. (2018). A call for clarity in reporting BLEU scores. In: Proceedings of the Third Conference on Machine Translation: Research Papers, Association for Computational Linguistics, Belgium, Brussels, pp 186–191, <https://www.aclweb.org/anthology/W18-6319>
- Slayden, G., Hwang, M. Y., & Schwartz, L. (2010). Thai sentence-breaking for large-scale smt. In: Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing, pp. 8–16
- Sornlertlamvanich, V., Charoenporn, T., & Isahara, H. (1997). Orchid: Thai part-of-speech tagged corpus. National Electronics and Computer Technology Center Technical Report pp. 5–19
- Tangsirirat, N., Suchato, A., Punyabukkana, P., & Wutiwiwatchai, C. (2013). Contextual behaviour features and grammar rules for thai sentence-breaking. *2013 10th International Conference on Electrical Engineering/Electronics* (pp. 1–4). Telecommunications and Information Technology, IEEE: Computer.
- Thompson, B., & Koehn, P. (2019). Vecalign: Improved sentence alignment in linear time and space. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, pp. 1342–1348, <https://doi.org/10.18653/v1/D19-1136>, <https://www.aclweb.org/anthology/D19-1136>
- Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. In: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), European Language Resources Association (ELRA), Istanbul, Turkey, pp. 2214–2218, http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf
- Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., & Trón, V. (2007). Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series, 4* (292), 247.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, AN., Kaiser, L., Polosukhin, I. (2017). Attention is all you need. CoRR [arXiv:abs/1706.03762](https://arxiv.org/abs/1706.03762)
- Yang, Y., Cer, D. M., Ahmad, A., Guo, M., Law, J., Constant, N., Ábrego, G. H., Yuan, S., Tar, C., Sung, YH., Strophe, B., & Kurzweil, R. (2019). Multilingual universal sentence encoder for semantic retrieval. [ArXiv:abs/1907.04307](https://arxiv.org/abs/1907.04307)
- Zaidan, O. (2012). Crowdsourcing annotation for machine learning in natural language processing tasks
- Ziemski, M., Junczys-Dowmunt, M., & Poulliquen, B. (2016). The united nations parallel corpus v1.0. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation

(LREC'16), European Language Resources Association (ELRA), Portorož, Slovenia, pp. 3530–3534, <https://www.aclweb.org/anthology/L16-1561>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.