Handling Cross- and Out-of-Domain Samples in Thai Word Segmentation

Peerat Limkonchotiwat
School of Information Science and
Technology, VISTEC, Thailand
Peerat.l_s19@vistec.ac.th

Wannaphong Phatthiyaphaibun
Faculty of Interdisciplinary Studies,
Khon Kaen University, Thailand
wannaphong@kkumail.com

Raheem Sarwar
RGCL, University of Wolverhampton,
United Kingdom
R.Sarwar4@wlv.ac.uk

Ekapol Chuangsuwanich
Department of Computer Engineering,
Chulalongkorn University, Thailand
ekapolc@cp.eng.chula.ac.th

Sarana Nutanong
School of Information Science and
Technology, VISTEC, Thailand
snutanon@vistec.ac.th

Abstract

While word segmentation is a solved problem in many languages, it is still a challenge in continuous-script or low-resource languages. Like other NLP tasks, word segmentation is domain-dependent, which can be a challenge in low-resource languages like Thai and Urdu since there can be domains with insufficient data. This investigation proposes a new solution to adapt an existing domain-generic model to a target domain, as well as a data augmentation technique to combat the low-resource problems. In addition to domain adaptation, we also propose a framework to handle out-of-domain inputs using an ensemble of domain-specific models called *Multi-Domain Ensemble* (MDE). To assess the effectiveness of the proposed solutions, we conducted extensive experiments on domain adaptation and out-of-domain scenarios. Moreover, we also proposed a multiple task dataset for Thai text processing, including word segmentation. For domain adaptation, we compared our solution to the state-of-the-art Thai word segmentation (TWS) method and obtained improvements from 93.47% to 98.48% at the character level and 84.03% to 96.75% at the word level. For out-of-domain scenarios, our MDE method significantly outperformed the state-of-the-art TWS and multi-criteria methods. Furthermore, to demonstrate our method's generalizability, we also applied our MDE framework to other languages, namely Chinese, Japanese, and Urdu, and obtained improvements similar to Thai's.

1 Introduction

Word segmentation (WS) is a crucial upstream process for most *natural language processing* (NLP) tasks such as *named entity recognition* (NER), *machine translation* (MT), and *part-of-speech tagging* (POS). Nguyen et al. (2017) showed POS performance increased from 87% to 93% when the WS was improved. WS can also enhance the performance of MT, such as the work done by Chang et al. (2008) for Chinese-English MT.

While word segmentation is considered a solved problem in many languages, the task is still a challenge in continuous-script languages. A great number of writing systems have no word boundary, e.g., Thai, Chinese, and Japanese. Deep learning has been effective in performing WS in these languages. However, it requires a large amount of training data to construct a reliable model, which can be a limitation for low-resource languages like Thai and Urdu. Furthermore, like other NLP tasks, word segmentation is domain-dependent (Fu et al., 2020). To handle a variety of data domains, there should be a substantial amount of data for each of them, exacerbating the low-resource problem. To make the matter worse, we may also need to handle input from a completely unseen domain.

In this paper, we propose a framework to address two domain dependency problems: (i) how to effectively construct a WS model to handle input from a given domain in a data-poor setting; (ii) how to effectively handle out-of-domain input. To address the first problem, we propose a new domain adaptation solution based on the concept of stacked ensemble (SE) learning (Limkonchotiwat et al., 2020) and data augmentation. To handle out-of-domain input, we use an ensemble of domain-specific models to produce predictive results.

The crux of our proposed method lies in the following technical contributions:

- We introduce multiple deep learning models following the concept of SE to construct domain-specific models that obtain better performance than the original SE and existing techniques in domain-adaptation problems. We call this technique *Deep Stacked Ensemble* (DSE).
- To make sure that each domain has sufficient data to build an accurate model, we design a data augmentation approach which consists of two techniques to generate hard-to-segment and semi-hard-to-segment samples to help improve the performance based on *Masked Language Model* (MLM).
- We use multiple domain-specific models and a result aggregation module to form an ensemble learning framework addressing the out-of-domain problems. We call this method *Multi-Domain Ensemble (MDE).*
- Furthermore, we propose a multiple task dataset called "*VISTEC-TP-TH-2021*", a social media dataset for Thai text processing, annotated for four text processing tasks: word segmentation, named-entity boundary, and misspelling detection and correction.

To assess the effectiveness of our approach, we compare our method with competitors in domain adaptation and out-of-domain scenarios on Thai, Chinese, Japanese, and Urdu. Experimental results showed that DSE improved the performance of the state-of-the-art Thai word segmentation (TWS) from 93.47% and 84.03% to 96.67% and 91.51% at character and word levels in domain adaptation settings. With the proposed data augmentation approach, our domain-specific model has improved even further at both character and word levels. For out-of-domain scenarios, our MDE framework outperformed the state-of-the-art TWS and multi-criteria baseline at character and word levels. Moreover, we applied our framework to Chinese, Japanese, and Urdu which resulted in improvement showing the applicability of our method to other languages. We make our code available at: `github.com/mrpeerat/OSKut`

2 Related Work

In this section, we discuss literature related to our investigation, namely *ensemble learning*, *domain adaption*, and *data augmentation*.

Ensemble Learning. Recently, considerable research attention has been dedicated to applying ensemble learning to boost the performance obtained from individual models (Sikdar and Gambäck, 2017; Chen et al., 2020a; Kuwabara et al., 2020) and to introduce previously ignored features for ensemble models such as provenance information in slot filtering (Viswanathan et al., 2015).

Several studies have used ensemble methods to boost the accuracy in WS. For example, Liu and Lin (2014) proposed a probabilistic ensemble learning framework using multiple weak word segmenters to form a strong segmenter. Moreover, Min et al. (2015) proposed an ensemble learning model to address the word segmentation and Part-of-Speech tagging problems by combining both discriminative and generative methods.

Domain Adaptation. Several WS studies proposed techniques to adapt the data distribution from one domain to another (Zhang et al., 2013; Ding et al., 2020). Another popular approach is to add new features or change network architectures of the target model (Monroe et al., 2014; Liu et al., 2014; Bao et al., 2017; Huang et al., 2020).

Ding et al. (2020) presented a semi-supervised approach for performing Chinese WS on a new domain by using adversarial training to help learn the difference between the source and target domain. Recently, Limkonchotiwat et al. (2020) proposed a filter-and-refine solution based on the stacked ensemble (SE) to convert a base model to a target domain. The SE consists of a domain-generic base model and a domain-specific model that analyzes the output of the domain-generic model and revises the segmentation. The method achieved similar performance to traditional transfer learning methods while requiring no access to the domain-generic model weights.

Data Augmentation and Self-Supervised learning. Word segmentation for low-resource languages is a challenging task due to the data limitation. Most Thai WS models report below 90% accuracy in domain-adaptation settings (Kittinaradorn et al., 2019; Chormai et al., 2020). Many researchers proposed data augmentation methods for Asian languages to increase the performance of WS models by using existing models' output as input to new models such as synthetic data, entropy parser, and character embedding (Zheng et al., 2018; Wang et al., 2019; Fung et al., 2004).

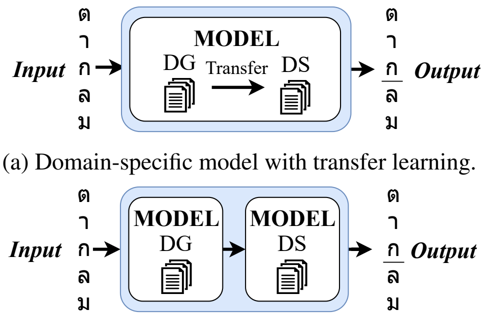
With the advent of large language models (Devlin et al., 2018; Yang et al., 2019; Brown et al., 2020), we have been witnessing an explosion in self-supervised learning techniques. Data augmentation methods such as the *Masked Language Model* (MLM) using BERT (Devlin et al., 2018) allow us to generate new sentences that are similar to real data by randomly selecting words in a sentence to replace them with new words (Chen et al., 2020b; Liao et al., 2020). Yavuz et al. (2020) proposed MaskAugment, a controllable mechanism and augmentation method that used a pre-trained BERT model to replace words in a sentence. The method is used in an unsupervised teacher-student framework to improve domain adaptation for dialog act task. Furthermore, Li et al. (2020) proposed a MLM-based augmentation method that could also preserve the underlying labels of the sentence in the aspect term extraction task.

Out-Of-Domain Scenarios. While domain adaptation presents a useful paradigm to adjust an existing model to a target domain, it is impracticable to anticipate all different input types in advance. Hence, the ability to handle samples from unseen domains (i.e., out-of-domain samples) is critical to the solution's performance. For example, Wagner et al. (2020) proposed utilizing treebank vectors and a method to interpolate a prediction from existing treebank vectors to handle out-of-domain input samples. Ng et al. (2020) proposed a solution utilizing data augmentation to generate training samples to diversify the training set so that the model can handle out-of-domain samples better.

Discussion. For domain adaptation, an ensemble learning method such as SE (Limkonchotiwat et al., 2020) provides a flexible framework for adapting any base model to a target domain. We hypothesize that we can improve the accuracy of SE by introducing a deep learning architecture at the domain-specific part. However, this adjustment would require a larger amount of data for each domain than the original SE method which uses a traditional Conditional Random Field (CRF) model (Lafferty et al., 2001). To tackle this problem, data-augmentation presents an avenue to address the data requirements. Regarding out-of-domain scenarios, we hypothesize that an ensemble of domain-specific models can be used to boost the accuracy of out-of-domain situations. This is the first WS work to address this problem without using any out-of-domain data.

3 Methodology

In this section, we present the overview of domain adaptation in Section 3.1. We present our domain adaptation solution based on the concept of stacked ensemble (SE) learning (Limkonchotiwat et al., 2020) and an MLM-based data augmentation method in Section 3.2. Section 3.3 presents how multiple domain-specific models can work as an ensemble to support out-of-domain scenarios.



(a) Domain-specific model with transfer learning.

(b) Domain-specific model with stacked ensemble.

Figure 1: Comparison between transfer Learning (TL) and SE to build domain-specific (DS) models. For domain-specific with transfer learning, there is one model that is first trained on domain-generic (DG) data then domain-specific data. For SE there are two models. The first is a domain-generic model that feeds initial prediction results to a second domain-specific model.

3.1 Overview of Domain Adaptation

A popular method to construct a *domain-specific* (DS) model is to adapt from a *domain-generic* (DG) base model using transfer learning as shown in Figure 1a. Stacked ensemble learning presents an alternative when making changes to the base model is impossible. Unlike transfer learning, the SE model consists of two parts: a domain-generic model and a domain-specific model as shown in Figure 1b. The domain-specific model takes the output predictions from the domain-generic model to make better predictions on the target domain. Before feeding into the domain-specific model, there is also a filter-and-refine stage where only uncertain predictions are re-visited by the domain-specific model. Only the domain-specific model is trained on the target data, while the base model is left untouched.

The main advantages of SE over TL are as follows: (i) the architecture of the domain-specific model can be selected independently of the existing domain-generic one; (ii) it is able to handle models where we cannot adjust their weights, i.e., black boxes. Consequently, we adopt SE as our approach to tackling the domain adaptation problem.

3.2 Deep Stacked Ensemble (DSE)

As stated earlier, SE allows us to introduce a new architecture to handle domain-specific input. To exploit this advantage, we introduce the *Bidirectional Long Short-Term Memory* (Bi-LSTM) with *Attention mechanism* to the current state-of-the-art TWS architecture (Kittinaradorn et al., 2019). We call our proposed domain adaptation method *Deep Stacked Ensemble* (DSE).

Figure 2 shows the structure of the domain specific part of our solution. There are three main kinds of features. A character n-gram is passed through a CNN following Kittinaradorn et al. (2019) to create an embedding vector (shown in blue). A character type n-gram which indicates whether a character is either a vowel, digit, special character, or an English character, is turned into an embedding vector (shown in red). Lastly, we use probability and entropy values from the domain-generic model, which indicates whether a character is a start or end of a word or not in a dictionary, as the additional features (colored as green). We then concatenate all of the embeddings and feed them to the Bi-LSTM layer (Hochreiter and Schmidhuber, 1997; Ma et al., 2018).

The Attention model is connected to the Bi-LSTM output layer for improved accuracy because the attention mechanism is effective at capturing long-range dependencies (Duan and Zhao, 2020). The attention layer is followed by a fully connected network that ends with a single sigmoid output for Thai and Chinese (boundary or not) and a softmax output for Japanese and Urdu (the beginning, middle, or end of a word, or a word with a single character).

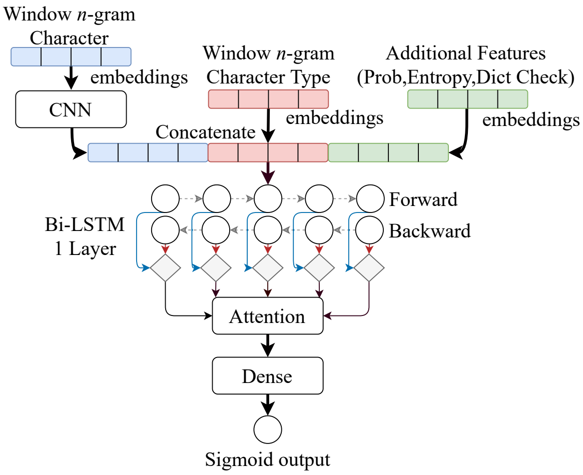


Figure 2: The domain-specific model of DSE.

Ablation studies, results given in Table 11, show that each component in the domain-specific model improves the performance incrementally. Unlike the original SE that relies only on the CRF as the domain-specific model, our deep learning approach to construct the domain-specific model can capture intricate WS patterns in the domain better than the original SE and transfer learning method.

However, unlike deep learning approaches, the classical machine learning approach, i.e., CRF, does not require a large amount of training data. To handle this problem, we propose the data augmentation technique at the character level. This can increase the amount of training data and thus improves the performance significantly.

Data Augmentation The main advantage of using a separate model for each domain is the ability to handle contradicting segmentation conditions from different domains (Fu et al., 2020). However, this approach requires a substantial amount of data in each domain as stated earlier. To mitigate this problem, we also propose two data augmentation methods based on the *Masked Language Model* (MLM) WangchanBERTa (Lowphansirikul et al., 2021) trained on Thai Wikipedia Dump. As shown in Figure 3, we mask words based on the output of the domain-generic model. The output posteriors from the model are used to compute the character-level entropy values. Then, the values are summed together to represent the score for each word. We select the words with the highest scores to mask in order to perform data augmentation. This is done to favor long words, since long words are harder to segment. We select the the top-k words to mask and replace them (substitution) using MLM. This a pretrained process to ensure the generation of hard-to-segment sentences. We also introduce semi-hard-to-segment samples by preferring word insertion after the word (rather than substitution). The same MLM is used to perform next word prediction instead of masked prediction. The ratio between hard-to-segment and semi-hard-to-segment is 80:20. This is found via grid search (see Table 13).

The insertion method gives the best performance compared with other semi-hard-to-segment generation methods (see the results in Table 13). The entropy selection method, compared with competitive selection methods in Table 12, shows that our method has the best performance for all Top-k selection and average scores.

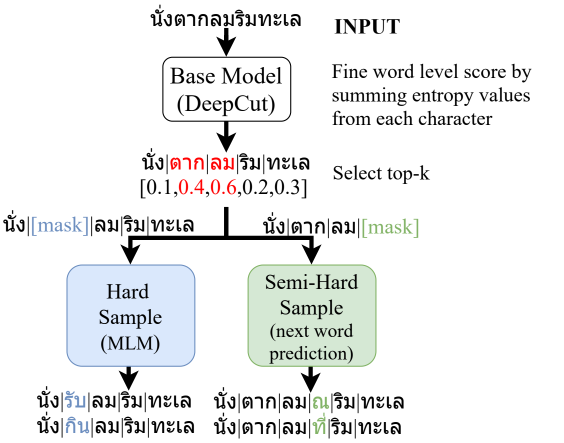


Figure 3: Overview of our data augmentation pipeline.

3.3 Muti-Domain Ensemble (MDE)

It is unrealistic to expect that the training and test distributions always match. Getting new training data for the out-of-domain scenarios can be expensive and time consuming (Ng et al., 2020; Liu et al., 2019). In such cases, transfer learning or the previously described DSE method are not sufficient.

We propose a framework, which utilizes an ensemble of domain-specific models to handle out-of-domain samples, called *Muti-Domain Ensemble (MDE)*. Figure 4 presents the structure of MDE.

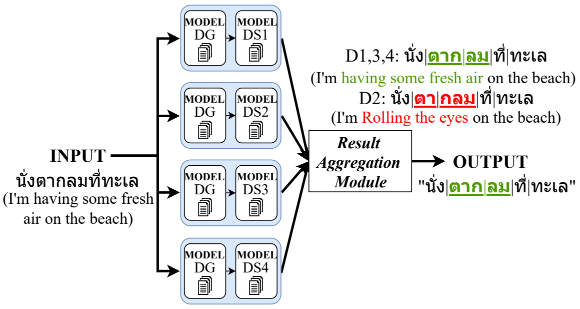

INPUT
นั่งตากลมที่ทะเล
(I'm having some fresh air on the beach)
MODEL DG
MODEL DS1
MODEL DS2
MODEL DS3
MODEL DS4
D1,3,4: นั่ง|ตาก|ลม|ที่|ทะเล
(I'm having some fresh air on the beach)
D2: นั่ง|ตา|กลม|ที่|ทะเล
(I'm Rolling the eyes on the beach)
Result Aggregation Module
OUTPUT
"นั่ง|ตาก|ลม|ที่|ทะเล"

Figure 4: *Multi-Domain Ensemble (MDE)*.

The framework consists of multiple word segmentation models, where each model is adapted to a specific target domain (except for the out-of-domain data) using the method described in Section 3.2. Results from multiple word segmentation models are combined using a result aggregator to form the final prediction. In this investigation, we formulate two result aggregation strategies as follows. (i) We compute a simple average of the posterior output from each model. Then, we predict the class that has the highest probability: this is a basic method for ensemble modeling (*Avg*); (ii) We calculate the entropy from each model based on their posterior distribution output. We then choose the prediction of the model with the lowest entropy and we call it *Min Entropy (ME)*.

The results of the MDE framework and aggregation module given in Tables 5 and 10 show that the entropy method performs better than the basic method and improves the performance significantly over other models in out-of-domain scenarios.

4 Performance Evaluations on Thai Word Segmentation

In this section, we report results from experimental studies on four *Thai word segmentation* (TWS) benchmark datasets. The studies are organized as follows. (i) we compare our method with competitive methods on domain adaptation; (ii) we show the effect of the data augmentation technique on domain adaptation; (iii) we report the results on out-of-domain setups; (iv) we show the effect of WS in downstream tasks.

Note that experimental studies on Chinese, Urdu, and Japanese are presented in Section 5.

4.1 Experimental Setup

Competitive Methods. We evaluate our proposed solution against two state-of-the-art methods namely DeepCut (DC) (Kittinaradorn et al., 2019) and AttaCut (AC) (Chormai et al., 2020). These methods are based on the *Convolutional Neural Network* (CNN) and trained on a generic corpus (BEST2009 (Boriboon et al., 2009)). For domain adaptation experiments, we also applied the concept of *Transfer Learning* (TL) to adapt DC and AC to the target corpora, and we call these adaptations TL-DC and TL-AC, respectively. Similarly, for the Stacked Ensemble Filter-and-Refine (SEFR) method (Limkonchotiwat et al., 2020), we created two variants, SE-DC and SE-AC, using DC and AC as the base model, respectively. For the evaluation of our method, *Deep Stack Ensemble* (DSE), we followed the same principle and created two variants DSE-DC and DSE-AC based on DC and AC, respectively.

Evaluation Metrics. We use F1 score as the evaluation metric for the TWS task at character and word levels to avoid the overestimation of TWS (Chormai et al., 2020; Limkonchotiwat et al., 2020).

Parameter Settings. In these experiments, we used grid search on 4 parameters including Bi-LSTM nodes, attention nodes, optimizer, and top-k inside the domain-specific model. We started the learning rate at 0.01 on an optimizer. For every 10 steps where the loss did not decrease, the learning rate was multiplied by a factor of 0.1. We set the number of training epochs to 300 with an option of early stopping. For the CNN layer and character embedding settings, we followed Kittinaradorn et al. (2019). We tuned the top-k value of the filtering system in a domain-specific model to be the same as the original SE. For the top-k value in the out-of-domain scenarios, we used the same k for all domain-specific models in the domain adaptation settings. Lastly, we tuned all of the parameters by using 10% of training data of the target domain. The hyper-parameters and their values are given in Table 1.

Hyper-parameters	Values for grid search
Optimizer	[Adam,RMSprop]
Learning Rate	0.01
Bi-LSTM nodes	[128, 160, 192, 224, 256]
Attention nodes	[32, 64, 96, 128, 160]
Top-k	[1-100]

Table 1: Hyper-parameters list.

4.2 Datasets

Benchmark Datasets. Our benchmark corpora can be seen in Table 2. They vary in domain and size from very small (Wisesight (Suriyawongkul et al., 2019) social media domain), moderate (TNHC (Sawatphol, 2019) classical literature), and large (LST20 (Boonkwan et al., 2020) generic).

Lang	Corpora	# Sentences	# Words
TH	Wisesight	1K [0.16K]	22K [3.9K]
TH	TNHC	13K [7K]	374K [239K]
TH	LST20	63.3K [5.2K]	2.7M [207K]
TH	VISTEC	39.8K [9.9K]	2.7M [690K]

Table 2: TWS corpora (# Training [# Testing]).

New Dataset. Due to social media data being underrepresented and difficult (Medina Serrano et al., 2020; Benton et al., 2017), it is challenging to improve the performance of models with only 997 training sentences. Most TWS models (Kittinaradorn et al., 2019; Chormai et al., 2020) performed under 82% in out-of-domain social media scenarios (Wisesight). To address this problem, we introduce a new dataset called "*VISTEC-TP-TH-2021*[1]" (VISTEC), which consists of 49,997 text samples from Twitter (2017-2019). VISTEC corpus contains 49,997 sentences with 3.39M words where the collection was manually annotated by linguists on four tasks, namely word segmentation, misspelling detection and correction, and named entity recognition. In the data collection process, we focused on the longest sentences to create a more challenging dataset due to the fact that long sentences made the model's performance decrease significantly compared with short sentences in the same domain (Section 4.3). The Out-of-Vocabulary rate on the test set is 13.65%.

We followed Boriboon et al. (2009) for the word and named entity tasks annotation guideline. We also included new guidelines about word editing criteria for misspelt words such as words used on the internet (Netspeak), transliterated loanwords, abbreviations, and shortened words, by using the Royal Institute Thai dictionary. We compared our dataset to the biggest Thai social media dictionary (Horsuwan et al., 2020) and found 79K words that did not appear in the dictionary.

4.3 Domain Adaptation

Without Data Augmentation. We evaluate the performance of our domain-specific model against competitive methods in four TWS benchmark corpora, WS160, TNHC, LST20, and VISTEC. The experimental results are given in Table 3. The competitive methods are defined in Section 4.1.

The DSE-DC (DeepCut) outperformed the strongest base model, DC, by 3.2% and 7.2% on WS160, 6.23% and 13.74% on TNHC, 4.41% and 10.18% on LST20, and 4.59% and 11.13% on VISTEC at character and word levels, respectively. Our domain-specific model also outperformed the original SE by 2.16% and 6.46% on SE-DC and 1.87% and 4.88% on SE-AC (AttaCut) at character and word levels on all setups. More importantly, our domain-specific model outperformed TL (transfer learning) methods showing the strength of our DSE model.

As expected, the newly constructed TWS social media dataset (VISTEC) shows that even TL-DC performed below 91% at word level, a large drop from the 96% achieved in the generic domain LST20 corpus. Also, the VISTEC dataset creates a new challenge for the social media domain. Comparing the WS160 and VISTEC datasets, the AC's performance decreased from 93.5% to 91.47% and 84.04% to 79.30% at the word level and the character level, respectively.

With Data Augmentation. In this experiment, we show the effect of the data augmentation in domain adaptation settings for different amounts of

[1] https://github.com/mrpeerat/OSKut/tree/main/VISTEC-TP-TH-2021

Method	WS160		TNHC		LST20		VISTEC	
	Char	Word	Char	Word	Char	Word	Char	Word
DC	93.47	84.03	89.48	75.40	94.60	87.15	92.77	81.78
AC	93.50	84.04	88.82	73.71	95.24	87.21	91.47	79.31
TL-DC	96.30	90.60	95.43	88.60	98.63	96.30	96.78	90.99
TL-AC	94.10	85.00	90.57	77.54	98.04	94.77	95.47	89.27
SE-DC	95.20	86.90	95.20	84.10	94.96	87.72	94.76	86.33
SE-AC	94.50	85.60	93.70	83.90	96.30	89.87	93.86	84.43
DSE-DC	**96.67**	**91.51**	**95.71**	**89.14**	**99.01**	**97.33**	**97.36**	**92.91**
DSE-AC	94.57	86.24	95.51	88.52	98.46	95.79	97.31	92.78

Table 3: Performance comparison on TWS in domain adaptation settings. TL, SE, and DSE models used target domain data besides BEST2009. DC = DeepCut, AC = AttaCut, TL = transfer learning, SE= stacked ensemble, and DSE = deep stacked ensemble.

adaptation data. We report the findings of the data augmentation process on 2 corpora, i.e., Wisesight (social media domain) which is the smallest corpus and LST20 which is the largest generic domain corpus LST20 (see Table 2). We fixed the top-k value in the data augmentation step at 60% and 10% of the segmentation predictions of the Wisesight and LST20 corpora, respectively. This value is found via grid search (see Table 12). We then use these augmented data with TL, SE, and DSE.

As shown in Table 4, the data augmentation process can improve the performance in the small corpus, i.e., Wisesight (WS160). DSE-DC (DeepCut) outperformed the base model by 5.01% and 12.72% at character and word levels. Also, DSE-DC outperformed TL-DC by 1.39% and 3.36% at the character and word levels respectively.

Method	WS160 (F1)		LST20 (F1)	
	Char	Word	Char	Word
DC	93.47	84.03	94.60	87.15
AC	93.50	84.04	95.24	87.21
TL-DC	97.69	93.96	98.11	94.00
TL-AC	97.59	94.57	97.45	93.25
SE-DC	95.08	86.37	96.47	90.40
SE-AC	94.66	86.36	96.28	89.77
DSE-DC	98.48	96.75	**98.67**	**97.03**
DSE-AC	**98.60**	**96.99**	98.61	96.18

Table 4: Performance on augmented WS160 and LST20.

However, since LST20 is sufficiently large, the augmentation did not produce performance improvement with respect to the model constructed using the original data only.

4.4 The Effect of Data Augmentation in Insufficient Data Scenarios

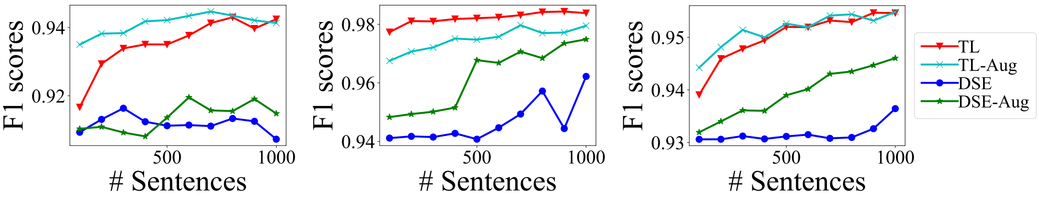
In this experiment, we evaluated the transfer learning (TL) and our method (DSE) trained on a varying numbers of sentences ranging from 100 to 1000 on the large datasets TNHC, LST20, and VISTEC to show the effectiveness of data augmentation in the insufficient data scenarios. As can be seen from Figure 5, the data augmentation improved the performance by 0.77% on average for TNHC, 1.55% for LST20, and 0.19% for VISTEC using DSE on the proposed data augmentation technique. Also, the transfer learning F1 performance is improved by 0.14% on average for TNHC and 0.57% for VISTEC. However, the performance of transfer learning on the LST20 data augmentation technique did not improve on this method as the baseline model (DeepCut) was trained on the same domain as the LST20 corpus. The performance of transfer learning in this setting is similar to the LST20 transfer learning model in Table 3.

The results of our method in insufficient data scenarios show that we improved the performance using the proposed data augmentation method when the original data is insufficient. Also, the best number of sentences for the augmentation technique in transfer learning is between 100 to 500 sentences and for our method is 500 to 1,000 sentences.

4.5 Experiments on Out-of-Domain Scenarios

In this experiment, we evaluated our *Multi-Domain Ensemble* (MDE) framework against two methods namely, DC trained on BEST2009 and *Multi-Criteria* (MC). MC is a multi-task model which learns multiple segmentation criteria from different domains jointly use shared layers (Chen et al., 2017). For MC and MDE, the target domains were left out from the training and the models are trained on the remaining domain.

As shown in Table 5, the performance improvements on Wisesight and TNHC were statistically significant (P<0.001 using McNemar's test) compared with MDE-ME and DC. Moreover, in comparison to DC, the performance improvement provided by MDE-Avg was also statistically significant on TNHC. As a result of MDE framework, we improved the performance from the base model (DC) at character and word level by 1.17% and 3.53% on WS160, 2.97% and 6.77% on TNHC, 0.26% and 0.42% on LST20, and 0.68% and 1.17% on VISTEC. Moreover, our MDE framework also outperformed the MC model in this experiment with significant results. In addition, the ME (Min Entropy) can improve the performance better than



(a) TNHC (b) LST20 (c) VISTEC

Figure 5: The effectiveness of Data Augmentation with limitation of training data with DSE and TL at Chracter-level F1.

the Avg except on TNHC.

Method	WS160		TNHC		LST20		VISTEC	
	Char	Word	Char	Word	Char	Word	Char	Word
DC	93.47	84.03	89.48	75.40	94.60	87.15	92.77	81.78
MC-Avg	88.59	72.30	84.96	72.36	87.55	66.52	89.67	70.38
MC-ME	91.59	84.80	85.13	72.63	91.10	80.70	90.81	77.13
MDE-Avg	93.85	84.64	**92.45**	**82.17**	94.81	87.54	93.17	82.71
MDE-ME	**94.64**	**87.56**	90.29	76.98	**94.86**	**87.57**	**93.45**	**83.53**

Table 5: Out-of-domain experimental results when the base model of MDE is DC (DeepCut), ME = Min Entropy.

As mentioned earlier, word segmentation is a domain-dependent task and we cannot expect the input to always be in domain. A model that can robustly handle the out-of-domain scenarios is desirable. Even with the improvement gained by our proposed solution, the gap between out-of-domain and domain adaptation is still large, showing potential for further investigation. In the next experiment, we show the effect of the data augmentation on downstream tasks.

4.6 The Effect of Word Segmentation and Data Augmentation on Downstream Tasks

Previously, we showed the proposed data augmentation improved the performance of TWS in the domain adaptation settings. In this experiment, we applied TWS to downstream tasks such as named entity recognition (NER), text classification, and sentiment analysis compared with the TWS base model (DC and AC), TL, DSE, and DSE with augmented data. For the text classification experiments, we use Wongnai corpus and Wisesight corpus for sentiment analysis. The exact model setting and evaluation metric follow Thai classification benchmark [2]. For the NER experiment, we used NCRF++ (Yang and Zhang, 2018) trained with data from Nutcha (2016)'s work. We trained our DSE and competitive methods (except the baseline model) on the Wisesight corpus to show the performance of the proposed augmentation technique.

The results are given in Table 6. When the downstream tasks are not dependent on WS performance, the results one similar i.e., text and sentiment classification tasks. On the other hand, when the downstream task is dependent on WS performance, i.e., NER, we can significantly improve the downstream task. For example, we improved the performance of DSE-DC from 93.47% to 96.67% at the character level, and when combined with data augmentation, increased the accuracy to 98.48%. As a result, the F1 score in the NER task increased from 63.46% to 72.27%.

Method	TC	SA	NER
DC	57.10	71.55	63.46
AC	57.20	71.66	72.98
TL-DC	57.26	72.38	61.70
TL-AC	56.72	71.61	64.28
DSE-DC	57.04	**72.63**	71.47
+ Augment	**58.01**	72.27	72.27
DSE-AC	57.01	72.35	66.60
+ Augment	56.99	72.41	**73.47**

Table 6: The effect of TWS data augmentation on downstream tasks. All models were trained on Wisesight.

5 Chinese, Urdu, and Japanese Word Segmentation

In this section, we demonstrate the generalizability of our method on Chinese word segmentation (CWS), Urdu word segmentation (UWS), and Japanese word segmentation (JWS). For Chinese and Urdu, we performed in-domain experiments showing the effectiveness of DSE over SE and other competitive baselines. For Japanese and Chinese, we show the effectiveness of DSE and MDE over baselines in domain adaptation and out-of-domain settings. The corpora used in these experiments are shown in Table 7. The exact setup and evaluation metrics follow those of Limkonchotiwat et al. (2020).

[2] https://github.com/PyThaiNLP/classification-benchmarks

Lang	Corpora	# Sentences	# Words
C	AS (Emerson, 2005)	636K [13K]	4.8M [110K]
C	CITYU (CI)	46K [1.1K]	1.2M [28K]
C	MSR (MS)	56K [3.5K]	1.4M [91K]
C	PKU (PK)	7.7K [1.1K]	371K [48K]
J	GSD (GS) (Asahara et al., 2018)	7K [0.5K]	159K [12K]
J	Modern (MO)	0.6K [0.16K]	11K [2.6K]
J	PUD (PU)	0.7K [0.19K]	19K [5K]
U	UCRF (UC) (Bin Zia et al., 2018)	3.5K [825]	90K [21K]

Table 7: WS corpora (# Training [# testing]) for Chinese (C), Japanese (J), and Urdu (U).

In-Domain Experiments on Chinese and Urdu. For Chinese, we used PyWordSeg (Chuang, 2019) trained on each of the four Chinese corpora as our baseline models (*BL*). We then trained SE and DSE models on top of each baseline model for each domain. For Urdu, we used the model and dataset provided by Bin Zia et al. (2018). The results are summarized in Table 8.

Method	Chinese				Urdu
	AS	CI	MS	PK	UC
BL	97.09	94.30	87.29	85.76	93.73
SE	97.51	96.13	93.82	93.55	93.90
DSE	**97.85**	**96.67**	**96.89**	**95.85**	**95.14**

Table 8: F1-scores on in-domain CWS and UWS tasks for each corpus. BL refers to the baseline chosen for each language.

Both stacked ensemble methods improve over the baseline models in all settings showing the potential of stacked ensemble in improving WS performance. Moreover, the proposed DSE outperforms the original SE (Limkonchotiwat et al., 2020) significantly for MSR, PKU, and UCRF (P<0.001). The largest performance improvement is over 10% on the PKU corpus.

Domain Adaptation on Japanese. As in the TWS experiments, DSE can also be used for domain adaptation by training the domain-specific portion on the target domain. For this JWS task, we used Nagisa (Ikeda, 2018) trained on *Balanced Corpus of Contemporary Written Japanese* (BCCWJ) (Maekawa et al.) corpora as the base model. The domain-specific part of the SE was trained on the target corpus to create an adapted model. Note that the Nagisa model released does not lend itself for transfer learning because the authors did not provide the model weights. From Table 9 SE and DSE improves significantly over the baseline showing the effectiveness of SE in situations when one cannot perform typical transfer learning.

Method	Japanese		
	GS	MO	PU
BL	87.10	78.80	87.10
SE	90.11	90.27	91.76
DSE	**92.36**	**90.65**	**91.89**

Table 9: F1-scores on domain adaptation JWS tasks for each corpus.

Out-of-Domain Experiments on Chinese and Japanese. Multiple models can form an MDE to provide robustness in out-of-domain scenarios. For Chinese, we used PyWordSeg trained on the AS corpus as the base model. The MDE included two domains (non-target) and was tested on the left-out target domain.

Method	Chinese			Japanese		
	CI	MS	PK	GS	MO	PU
BL	92.51	83.92	82.21	87.10	78.80	87.10
MDE-Avg	88.90	89.15	89.77	87.11	79.36	87.42
MDE-ME	**93.98**	**93.01**	**93.56**	**87.12**	**79.39**	**87.44**

Table 10: F1-scores on out-of-domain CWS and JWS experiments.

Table 10 summarizes the results of the out-of-domain experiments. The MDE provides a minimal improvement over the baseline on JWS. We hypothesize that this is because two out of the three corpora are too small to train a reliable model. However, on Chinese the MDE provides large gains over the baseline with the min entropy method performing better than the simple averaging method.

6 Concluding Remarks

This investigation presents a set of solutions to address two domain dependency problems: handling cross-domain and out-of-domain samples. Our key findings are as follows. First, we applied deep learning to the original stacked ensemble method and obtained a significant improvement. Second, we show that data augmentation is an effective method to combat the low-resource limitation in domain adaptation. Third, we can use an ensemble of domain-specific models to obtain a performance improvement over each domain-specific model acting alone. Finally, in addition to Thai, we can apply the same principle to Chinese, Japanese, and Urdu and obtain similar improvements. As future work, we plan to experiment with novel techniques, i.e., Transformer and contrastive learning.

References

Masayuki Asahara, Hiroshi Kanayama, Takaaki Tanaka, Yusuke Miyao, Sumire Uematsu, Shinsuke

Mori, Yuji Matsumoto, Mai Omura, and Yugo Murawaki. 2018. Universal Dependencies Version 2 for Japanese. In *LREC*.

Zuyi Bao, Si Li, Weiran Xu, and Sheng Gao. 2017. Neural regularized domain adaptation for Chinese word segmentation. In *Proceedings of the 9th SIGHAN Workshop on Chinese Language Processing*, pages 11–20, Taiwan. Association for Computational Linguistics.

Adrian Benton, Glen Coppersmith, and Mark Dredze. 2017. Ethical research protocols for social media health research. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 94–102, Valencia, Spain. Association for Computational Linguistics.

Haris Bin Zia, Agha Ali Raza, and Awais Athar. 2018. Urdu word segmentation using conditional random fields (CRFs). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2562–2569, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

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.

Monthika Boriboon, Kanyanut Kriengket, Patcharika Chootrakool, Sitthaa Phaholphinyo, Sumonmas Purodakananda, Tipraporn Thanakulwarapas, and Krit Kosawat. 2009. Best corpus development and analysis. In *2009 International Conference on Asian Language Processing*, pages 322–327. IEEE.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Pi-Chuan Chang, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese Word Segmentation for Machine Translation Performance. In *WMT@ACL*, pages 224–232.

Jun Chen, Xiaoya Dai, Quan Yuan, Chao Lu, and Haifeng Huang. 2020a. Towards interpretable clinical diagnosis with Bayesian network ensembles stacked on entity-aware CNNs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3143–3153, Online. Association for Computational Linguistics.

Luoxin Chen, Xinyue Liu, Weitong Ruan, and Jianhua Lu. 2020b. Enhance robustness of sequence labelling with masked adversarial training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 297–302, Online. Association for Computational Linguistics.

Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-criteria learning for Chinese word segmentation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1193–1203, Vancouver, Canada. Association for Computational Linguistics.

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*, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yung-Sung Chuang. 2019. Robust Chinese Word Segmentation with Contextualized Word Representations. *CoRR*, abs/1901.05816.

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

Ning Ding, Dingkun Long, Guangwei Xu, Muhua Zhu, Pengjun Xie, Xiaobin Wang, and Haitao Zheng. 2020. Coupling distant annotation and adversarial training for cross-domain Chinese word segmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6662–6671, Online. Association for Computational Linguistics.

Sufeng Duan and Hai Zhao. 2020. Attention is all you need for Chinese word segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3862–3872, Online. Association for Computational Linguistics.

T. Emerson. 2005. The second international chinese word segmentation bakeoff. In *SIGHAN@IJCNLP 2005*.

Jinlan Fu, Pengfei Liu, Qi Zhang, and Xuanjing Huang. 2020. RethinkCWS: Is Chinese word segmentation a solved task? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5676–5686, Online. Association for Computational Linguistics.

Pascale Fung, Grace Ngai, Yongsheng Yang, and Benfeng Chen. 2004. A maximum-entropy chinese parser augmented by transformation-based learning. *ACM Transactions on Asian Language Information Processing*, 3(2):159–168.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9:1735–80.

Thanapapas Horsuwan, Kasidis Kanwatchara, Peerapon Vateekul, and Boonserm Kijsirikul. 2020. A comparative study of pretrained language models on thai social text categorization. In *ACIIDS*.

Kaiyu Huang, Degen Huang, Zhuang Liu, and Fengran Mo. 2020. A joint multiple criteria model in transfer learning for cross-domain Chinese word segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3873–3882, Online. Association for Computational Linguistics.

Taishi Ikeda. 2018. nagisa: A Japanese tokenizer based on recurrent neural networks. https://github.com/taishi-i/nagisa.

Rakpong Kittinaradorn, Korakot Chaovavanich, Titipat Achakulvisut, Kittinan Srithaworn, Pattarawat Chormai, Chanwit Kaewkasi, Tulakan Ruangrong, and Krichkorn Oparad. 2019. DeepCut: A Thai word tokenization library using Deep Neural Network.

Ryosuke Kuwabara, Jun Suzuki, and Hideki Nakayama. 2020. Single model ensemble using pseudo-tags and distinct vectors. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3006–3013, Online. Association for Computational Linguistics.

John Lafferty, Andrew Mccallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. pages 282–289.

Kun Li, Chengbo Chen, Xiaojun Quan, Qing Ling, and Yan Song. 2020. Conditional augmentation for aspect term extraction via masked sequence-to-sequence generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7056–7066, Online. Association for Computational Linguistics.

Yi Liao, Xin Jiang, and Qun Liu. 2020. Probabilistically masked language model capable of autoregressive generation in arbitrary word order. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 263–274, Online. Association for Computational Linguistics.

Peerat Limkonchotiwat, Wannaphong Phatthiyaphaibun, Raheem Sarwar, Ekapol Chuangsuwanich, and Sarana Nutanong. 2020. Domain adaptation of Thai word segmentation models using stacked ensemble. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3841–3847, Online. Association for Computational Linguistics.

Junxin Liu, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2019. Neural chinese word segmentation with lexicon and unlabeled data via posterior regularization. In *The World Wide Web Conference*, WWW '19, page 3013–3019, New York, NY, USA. Association for Computing Machinery.

Wuying Liu and Li Lin. 2014. Probabilistic ensemble learning for vietnamese word segmentation. In *Proceedings of the 37th International ACM SIGIR Conference on Research Development in Information Retrieval*, SIGIR '14, page 931–934, New York, NY, USA. Association for Computing Machinery.

Yijia Liu, Yue Zhang, Wanxiang Che, Ting Liu, and Fan Wu. 2014. Domain adaptation for CRF-based Chinese word segmentation using free annotations. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 864–874, Doha, Qatar. Association for Computational Linguistics.

Lalita Lowphansirikul, Charin Polpanumas, Nawat Jantrakulchai, and Sarana Nutanong. 2021. Wangchanberta: Pretraining transformer-based thai language models.

Ji Ma, Kuzman Ganchev, and David Weiss. 2018. State-of-the-art Chinese word segmentation with Bi-LSTMs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4902–4908, Brussels, Belgium. Association for Computational Linguistics.

Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. Balanced corpus of contemporary written Japanese. *Lang. Resour. Evaluation*.

Juan Carlos Medina Serrano, Orestis Papakyriakopoulos, and Simon Hegelich. 2020. NLP-based feature extraction for the detection of COVID-19 misinformation videos on YouTube. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Kerui Min, Chenggang Ma, Tianmei Zhao, and Haiyan Li. 2015. Bosonnlp: An ensemble approach for word segmentation and pos tagging. In *Natural Language Processing and Chinese Computing*, pages 520–526, Cham. Springer International Publishing.

Will Monroe, Spence Green, and Christopher D. Manning. 2014. Word segmentation of informal Arabic with domain adaptation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 206–211, Baltimore, Maryland. Association for Computational Linguistics.

Nathan Ng, Kyunghyun Cho, and Marzyeh Ghassemi. 2020. SSMBA: Self-supervised manifold based data augmentation for improving out-of-domain robustness. In *Proceedings of the 2020 Conference on*

Empirical Methods in Natural Language Processing (EMNLP), pages 1268–1283, Online. Association for Computational Linguistics.

Dat Quoc Nguyen, Thanh Vu, Dai Quoc Nguyen, Mark Dras, and Mark Johnson. 2017. From word segmentation to POS tagging for Vietnamese. In *Proceedings of the Australasian Language Technology Association Workshop 2017*, pages 108–113, Brisbane, Australia.

Tirasaroj Nutcha. 2016. *A study of word sense discrimination in Thai using latent semantic analysis*, volume 25.

Jitkapat Sawatphol. 2019. Thai literature corpora. urlhttps://attapol.github.io/tlc.html.

Utpal Kumar Sikdar and Björn Gambäck. 2017. A feature-based ensemble approach to recognition of emerging and rare named entities. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 177–181, Copenhagen, Denmark. Association for Computational Linguistics.

Arthit Suriyawongkul, Pattarawat Chormai, Charin Polpanumas, and Ekapol Chuangsuwanich. 2019. Pythainlp/wisesight-sentiment: First release.

Vidhoon Viswanathan, Nazneen Fatema Rajani, Yinon Bentor, and Raymond Mooney. 2015. Stacked ensembles of information extractors for knowledge-base population. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 177–187, Beijing, China. Association for Computational Linguistics.

Joachim Wagner, James Barry, and Jennifer Foster. 2020. Treebank embedding vectors for out-of-domain dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8812–8818, Online. Association for Computational Linguistics.

Xiaobin Wang, Deng Cai, Linlin Li, Guangwei Xu, Hai Zhao, and Luo Si. 2019. Unsupervised learning helps supervised neural word segmentation. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7200–7207. AAAI Press.

Jie Yang and Yue Zhang. 2018. NCRF++: An open-source neural sequence labeling toolkit. In *Proceedings of ACL 2018, System Demonstrations*, pages 74–79, Melbourne, Australia. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32, pages 5753–5763. Curran Associates, Inc.

Semih Yavuz, Kazuma Hashimoto, Wenhao Liu, Nitish Shirish Keskar, Richard Socher, and Caiming Xiong. 2020. Simple data augmentation with the mask token improves domain adaptation for dialog act tagging. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5083–5089, Online. Association for Computational Linguistics.

Longkai Zhang, Houfeng Wang, Xu Sun, and Mairgup Mansur. 2013. Exploring representations from unlabeled data with co-training for Chinese word segmentation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 311–321, Seattle, Washington, USA. Association for Computational Linguistics.

R. Zheng, M. Li, J. He, J. Bi, and B. Wu. 2018. Segmentation-free multi-font printed manchu word recognition using deep convolutional features and data augmentation. In *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–6.

A Appendices

A.1 Ablation Studies

In this experiment, we show the effect of Bi-LSTM with attention mechanism on the performance of the TWS domain adaption problem with the help of an ablation study. This study was performed on the Wisesight corpus (test set) and used DC (DeepCut) as the base model. As can been seen from Table 11, each component significantly improved the performance from the base model.

System	F1 Char	F1 Word
Baseline	93.47	84.03
+ Additional Feature	94.27	85.39
+ Bi-LSTM	96.31	90.35
+ Attention	**96.67**	**91.51**

Table 11: TWS domain adaptation ablation studies.

A.2 How to Select a Word to Mask? Entropy vs Random vs Maximum Length Selections.

The Section 3.2 presented the way we select a word to augment and compared it against the traditional method i.e., random selection. In this study, we show the validation score of random, entropy, and maximum length selections in our data augmentation technique on the Wisesight corpus with DC base model by varying the k value from top-10% to 100% on the substitution method and fixed top-20% value for the insertion method.

The validation score results are given in Table 12, the best range top-k value for the data augmentation is 50% to 60%. The performance of entropy selection is better than competitive methods with reasonable results. Due to the fact that long words are harder to segment than short ones, the entropy selection method favors long words with a high uncertainty. Maximum length selection, gives a similar score with entropy selection due to the fact that high uncertainty score mostly comes from long words. Also, the best F1 score is obtained using top 60% not 100% as words in top-60% might have the most incorrect answers and bias from frequency word.

Top-k Entropy	ES		RS		MLS	
Select	char	word	char	word	char	word
10	99.74	99.29	99.50	98.67	99.71	99.13
20	99.82	99.50	99.66	99.11	99.81	99.49
30	99.80	99.45	99.82	99.57	99.79	99.54
40	99.77	99.42	99.81	99.59	99.81	99.55
50	**99.90**	99.75	99.79	99.46	99.84	99.64
60	**99.90**	**99.77**	99.84	99.56	99.87	99.70
70	99.79	99.44	99.87	99.67	99.79	99.44
80	99.76	99.41	99.87	99.70	99.76	99.41
90	99.80	99.45	99.80	99.55	99.76	99.46
100	99.86	99.66	99.85	99.70	99.86	99.66
AVG	**99.81**	**99.51**	99.78	99.46	99.80	99.50

Table 12: Performance comparison (F1) on entropy selection (ES) vs random selection (RS) VS maximum length selection (MLS) on validation scores.

A.3 Ablation Studies For Different Semi-Hard-Sample Procedures

Section 3.2 mentioned a competitive method to produce semi-hard-to-segment samples. We use the semi-hard-to-segment method with substitution by fixing the k value at top-60% for substitution method and we vary k in the range of 10% to 100% on the semi-hard-to-segment methods to show the performance of each method. We show the validation score on Wisesight (training data) with character and word levels, respectively. The results are presented in Table 13. As can be seen, the insertion method reports the best performance on every top-k entropy selection. The deletion method is inappropriate due to the fact that we might delete some information in the training data.

Top-k Entropy	Insertion		Deletion	
Select	char	word	char	word
10	99.79	99.54	99.80	99.61
20	**99.90**	**99.77**	99.77	99.40
30	**99.90**	99.65	99.75	99.32
40	99.81	99.67	99.79	99.39
50	99.85	99.69	99.64	98.96
60	99.86	99.62	99.54	98.71
70	99.88	99.66	98.93	97.00
80	99.84	99.67	98.97	97.15
90	**99.90**	99.66	97.84	94.11
100	99.88	99.62	93.84	84.31
AVG	**99.86**	**99.66**	98.79	96.80

Table 13: Performance comparison on insertion VS deletion to produce semi-hard-to-segment samples with WS160 (F1 validation score).

A.4 Error Analysis

We performed an error analysis on Wisesight (WS160) corpora for DC, SE-DC, and DSE-DC to investigate the improvement from the baselines as well as the benefits of our method in domain adaptation setups. We used the same setting as mentioned in Section 4.3. The samples presented here were randomly selected from the Wisesight validation set.

As shown in Figure 6, DSE-DC did better especially on compound words. However, all models still cannot properly handle the special character (+), since the character is rare in the WS160 corpus.

Actual: ไม่|รุ้|นะ|คับ|แพ้|น้ำหอม|หรอ|แบบ|นี้
DeepCut: ไม่|รุ้นะ|คับ|แพ้|น้ำ|หอม|หรอ|แบบ|นี้
SE-DeepCut: ไม่|รุ้นะ|คับ|แพ้|น้ำ|หอม|หรอ|แบบ|นี้
DSE-DeepCut: ไม่|รุ้|นะ|คับ|แพ้|น้ำหอม|หรอ|แบบ|นี้

Actual: แต่|เทียน่า|มัน|ขาย|ไม่|ดี| |นิสสัน|คง|เน้น|ทำ|อีโค|คา|เต็ม|ตัว|555+
DeepCut: แต่|เทีย|น่า|มัน|ขาย|ไม่|ดี| |นิสสัน|คง|เน้น|ทำ|อีโคคา|เต็มตัว|555|+
SE-DeepCut: แต่|เทีย|น่า|มัน|ขาย|ไม่|ดี| |นิสสัน|คง|เน้น|ทำ|อีโคคา|เต็มตัว|555|+
DSE-DeepCut: แต่|เทียน่า|มัน|ขาย|ไม่|ดี| |นิสสัน|คง|เน้น|ทำ|อีโค|คา|เต็ม|ตัว|555|+

Actual: อยาก|กิง|บาบีก้อน|บุฟ|อีก|อ่ะ| |คิดถึง|ที่|ปี|ที่|แล้ว|ไป|กิน|กะ|มึง|ง่ะ
DeepCut: อยาก|กิงบาบีก้อน|บุฟ|อีก|อ่ะ| |คิด|ถึง|ที่|ปี|ที่|แล้ว|ไป|กินกะ|มึงง่ะ
SE-DeepCut: อยาก|กิงบาบีก้อน|บุฟ|อีก|อ่ะ| |คิด|ถึง|ที่|ปี|ที่|แล้ว|ไป|กิน|กะ|มึงง่ะ
DSE-DeepCut: อยาก|กิง|บาบีก้อน|บุฟ|อีก|อ่ะ| |คิดถึง|ที่|ปี|ที่|แล้ว|ไป|กิน|กะ|มึง|ง่ะ

Figure 6: The example of segmentation results in WS160 validation dataset.