Combining Thai EDUs: Principle and Implementation

Chanatip Saetia **Supawat Taerungruang** **Tawunrat Chalothorn**

Kasikorn Labs

Kasikorn Business Technology Group (KBTG)

Nontaburi, Thailand

chanatip.sae@kbtg.tech, supawat.t@kbtg.tech, tawunrat.c@kbtg.tech

Abstract

Due to the lack of explicit end-of-sentence marker in Thai, Elementary Discourse Units (EDUs) are usually preferred over sentences as the basic linguistic units for processing Thai language. However, some segmented EDUs lack of structural or semantic information. To obtain a well-form unit, which represents a complete idea, this paper proposes combining EDUs with rhetorical relations selected depending on our proposed syntactic and semantic criteria. The combined EDUs can be then used without considering other parts of the text. Moreover, we also annotated data with the criteria. After that, we trained a deep learning model inspired by coreference resolution and dependency parsing models. As a result, our model achieves the F1 score of 82.72%.

1 Introduction

Generally, sentences have served as the basic linguistic units required by many tasks in NLP (e.g., text summarization and question answering) for processing long bodies of text (Mihalcea, 2004; Raiman & Miller, 2017; Van Lierde & Chow, 2019). Basic linguistic units must contain complete propositional content that represents a single piece of idea. However, in Thai language, the sentence cannot clearly specify the boundary (Intasaw & Aroonmanakun, 2013, p. 491; Lertpiya et al., 2018) since there is no explicit end-of-sentence marker like a period in English.

For this reason, prior works (Ketui et al., 2015; Singkul et al., 2019; Sinthupoun & Sornil, 2010; Sukvaree et al., 2004) have suggested to use the Elementary Discourse Unit (EDU), which is the basic unit in discourse based on Rhetorical Structure Theory (RST) (Mann & Thompson, 1988; Taboada & Mann, 2006), as a processing unit for Thai language. However, a single EDU, on its own, may not contain complete information to be understandable. Generally, EDUs are segmented based solely on syntactic criteria (Carlson & Marcu, 2001; Intasaw & Aroonmanakun, 2013; Ketui et al., 2012) and used alongside RST relation (which contains semantic and structural information) to represent complete discourse structure.
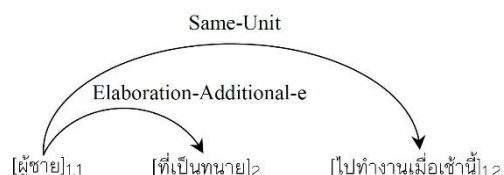

Figure 1. EDU that is separated due to embedded clause

To highlight that the EDU segmentation based on syntactic criteria creates individual EDUs with incomplete in structure and meaning, an example is shown in Figure 1. The figure illustrates RST d-tree (Morey et al., 2018) of EDU, which is separated by an embedded clause. A matrix clause [ผู้ชายไปทำงานเมื่อเช้านี้] 'A man went to work this morning', which should be one unit of EDU, is separated into two units, [ผู้ชาย] 'A man' and [ไปทำงานเมื่อเช้านี้] 'went to work this morning', because there is embedded clause

[*ที่เป็นทนาย*] 'who is a lawyer' modifying a noun [*ผู้ชาย*] 'A man' in matrix clause.

As such, combining incomplete EDUs is necessary for obtaining a complete idea that is to represent through a well-formed EDU.

This paper states the syntactic and semantic **criteria** for considering rhetorical relations used to combine EDUs into a well-formed EDU, which represents a complete idea. After that, rhetorical **relations** that correspond to those criteria are proposed.

We conducted experiments by building a dataset based on our proposed criteria and relations, which is then trained on a deep learning model. Our first experiment investigates methods to score the combination of EDU pairs. Since there is no prior work on this exact task, two methods from dependency parser task (Dozat & Manning, 2016) and coreference resolution task (Lee et al., 2017) were adapted on this task. In the second experiment, various EDU representations constructed from contextual word vectors are compared against one another. Lastly, we discuss how the result of our method covers the proposed rhetorical relations stated above.

The rest of this paper is structured as follows. In Section 2, the background knowledge related to this work is reviewed. The type of rhetorical relation to combine EDUs is explained in Section 3. While, in Section 4, we describe the architecture of our deep learning model. The dataset, implementation detail, and evaluation metrics are mentioned in Section 5. The results are shown in Section 6 and discussed in Section 7. Finally, Section 8 concludes the paper.

2 Theoretical Background

2.1 Elementary discourse unit

For processing to obtain single pieces of information from text, the text needs to be segmented into units that are related to each other. For this reason, Rhetorical structure theory (RST), a text organization theory, was proposed by Mann and Thompson (1988). RST is widely applied in text and discourse processing. The essence of this theory is to analyze relationships between subunits in the discourse, which is based on the intention of the messenger and the content of text. A tree diagram is then used to represent the relationships between subunits.

RST's theoretical concept was developed to be more practical by Carlson et al. (2001). In the discourse structure, there is the smallest unit that can convey complete content and meaning, called Elementary Discourse Unit (EDU). EDU can determine the nuclearity status of each unit and can also specify the type of rhetorical relation between them. After that, the criteria used to segment EDUs in different languages and sets of rhetorical relations were proposed in large numbers.

In this paper, the criteria for segmenting Thai EDUs that proposed by Intasaw and Aroonmanakun (2013) was applied.

2.2 The prior principle to combine discourse units.

Schauer (2000) states that not only clauses but sometimes prepositional phrases can also be considered as discourse units. To support this statement, three principles for including phrases as discourse units are proposed as follows.

First, consideration of the complement-status and adjunct-status of a clause or phrase is the principle that if any discourse unit has a status as a complement of another discourse unit, they will be combined to express the complete meaning. On the other hand, if any discourse unit has a status as an adjunct of another discourse unit, there is no need to combine them into a larger unit, since the main discourse unit is already meaningful.

Second, consideration of semantic specification of lexeme is to consider whether each word in discourse unit needs to rely on other linguistic units for expressing its lexical meaning. For example, the unit that indicates the agent, the patient, the instrument, the location, or the time frame. If the words in the discourse unit need to rely on another discourse unit in order to express their lexical meaning, those units will be combined.

Finally, consideration of discourse relation between units is the principle that the combined

discourse units always show the discourse relation between each other. These relations can be identified by the function of conjunctions at the beginning of the unit.

In this paper, the principles proposed by Schauer (2000) are applied as criteria to combine EDUs by considering the rhetorical relations between EDUs that are related to syntactic and semantic characteristics of text. The details on applying those criteria are presented in the next section.

3 Types of Rhetorical Relation to Combine

By applying the concepts from Schauer (2000), EDUs occurred in discourse structure can be considered as a clause or a phrase with the status of complement or adjunct. Besides that, four types of rhetorical relations only used to link between the matrix clause and unit with complement-status. The units with complement-status are necessary to express the complete meaning. If such units are omitted, the expressed meaning is inadequate. The criteria for combing EDUs are applied by using four types of rhetorical relations that are linked between units with complement-status. The details of each type are explained in the following section.

3.1 Attribution

The attribution relation is used to link between a clause containing a reporting verb or attribution verb, e.g. *พูด* 'speak', *รายงาน* 'report', *บอก* 'tell', *คิด* 'think', *สั่ง* 'order', and a clause containing the content of the reported message (Carlson & Marcu, 2001, p. 46).
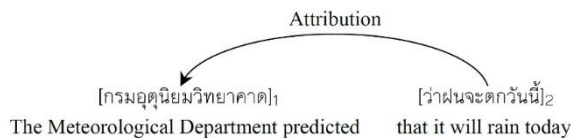

Figure 2. EDUs that linked by attribution relation

As shown in Figure 2, EDU_2 contains the content of the reporting verb *คาด* 'predict' in EDU_1. In terms of syntactic and semantic characteristics, EDU_2 is a complement clause of the main verb in EDU_1. If there is any part missing,

the incomplete meaning will be conveyed. For this reason, both EDUs must be combined.

3.2 Attribution-negative

The attribution-negative has properties similar to the attribution relation. The difference is that the attribution-negative is used for marking a negative attribution.
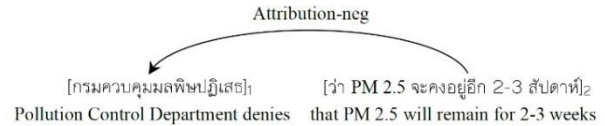

Figure 3. EDUs that linked by attribution-negative relation

Like the attribution relation, in Figure 3, EDU_2 contains the content which cannot be omitted of an attributive verb *ปฏิเสธ* 'deny' in EDU_1. But, in this case, the main verb appeared in EDU_1 is semantically negative. Therefore, the attribution-negative is applied.

3.3 Elaboration-object-attribute

Elaboration-object-attribute is a relation involving a clause, usually a postmodifier of a noun phrase in the matrix clause, that is required to give meaning to an animate or inanimate object (Carlson & Marcu, 2001, p. 54). In this case, omitting modifier clauses may result in incomplete meaning.
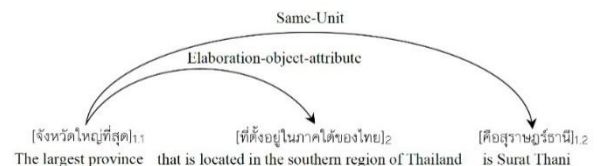

Figure 4. EDUs that linked by elaboration-object-attribute

Figure 4 shows that $EDU_{1.1}$ need EDU_2 as a modifier for a noun phrase *จังหวัดใหญ่ที่สุด* 'the largest province'. If EDU_2 is omitted, the combination of $EDU_{1.1}$ and $EDU_{1.2}$ will mean 'The largest province is Surat Thani', which is the meaning that causes misunderstandings. Therefore, to become a meaningful unit, EDU_2 is a complement that must always be combined to $EDU_{1.1}$.

3.4 Same-unit

Same-unit is pseudo-relation used as a device for linking two discontinuous text fragments that are

really a single EDU, but which are broken up by an embedded unit (Carlson & Marcu, 2001, p. 66).
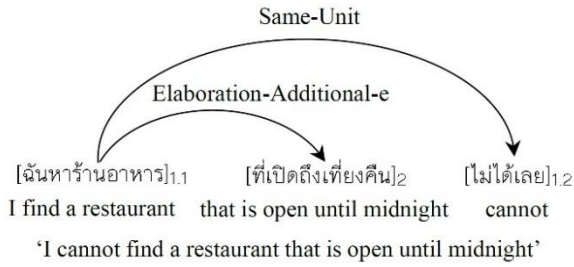

Figure 5. Single EDU that is separated due to embedded clause

Considering Figure 5, a matrix clause ฉันหาร้านอาหารไม่ได้เลย 'I cannot find a restaurant', which is a single EDU, is broken up into 2 units by an embedded clause with adjunct-status ที่เปิดถึงเที่ยงคืน 'that is open until midnight' modifying a noun ร้านอาหาร 'a restaurant' in a matrix clause. In this case, $EDU_{1.1}$ [ฉันหาร้านอาหาร] 'I find a restaurant' and $EDU_{1.2}$ [ไม่ได้เลย] 'cannot' have to be combined to represent the meaning as a single EDU.

The types of rhetorical relations presented in this section are used as theoretical backgrounds for implementing combining EDUs model. Details are discussed in the next section.

4 Model architecture

The architecture of our deep learning model for combining EDUs is presented in this section. The architecture, as shown in Figure 6, is separated into four parts: Contextual word representation, EDU representation, EDU combination scoring, and the training and inference process. The detail of each part is elaborated in the following subsections.

4.1 Contextual word representation

This module is responsible for converting a sequence of words $\vec{w} = [w_1, w_2, ..., w_N]$ and the corresponding part-of-speech tags (POS) $\vec{p} = [p_1, p_2, ..., p_N]$ into a sequence of contextual word vectors $H = [\vec{h_1}, \vec{h_2}, ..., \vec{h_N}]$ where N is the word sequence length, as illustrated in Figure 6. First, each word and its POS in the sequence is converted into embedding vectors $E = [\vec{e_1}, \vec{e_2}, ..., \vec{e_N}]$ where $\vec{e_t}$ is the concatenation of word embeddings and POS embeddings. After that, the sequence of concatenated embedding vectors is fed into Bidirectional Long-short memory network (Bi-LSTM) to create contextual word vectors H.

4.2 EDU representation

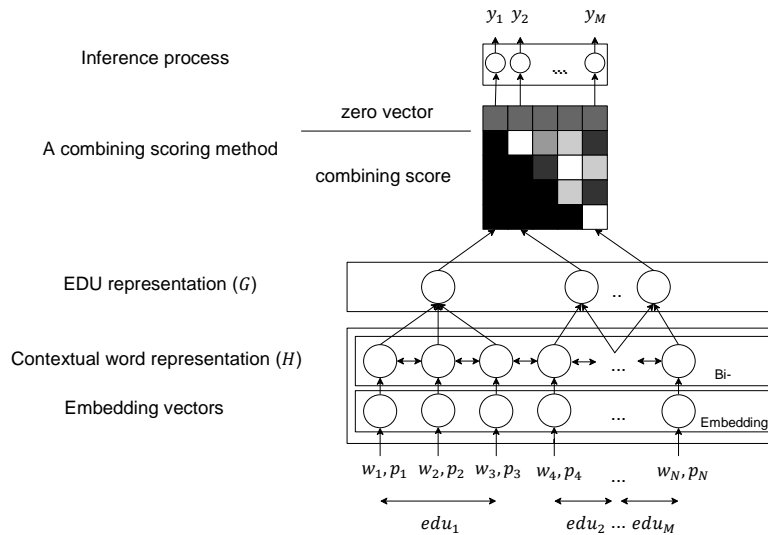This module aggregates the contextual word

Figure 6. Model Architecture

vectors H into EDU representation $G = [\vec{g_1}, \vec{g_2}, ..., \vec{g_M}]$ where M is the number of EDUs

and $\overrightarrow{g_t}$ represents one EDU (edu_t). For this module, we adopt two methods to create the EDU representation. These methods were proposed by Lee et al. (2017) to create span representation for performing coreference resolution. Therefore, in this module, each EDU vector $\overrightarrow{g_t}$ is concatenated from an end-point vector $\vec{g}_t^{(ep)}$ and a self-attention vector $\vec{g}_t^{(atten)}$, which are described in the following subsections.

End-point representation presents each edu_t by concatenating with three parts, as shown in Eq. 1. The beginning and end contextual word vectors ($\vec{h}_{begin(t)}, \vec{h}_{end(t)}$) are concatenated with the length of the EDU in words ($\phi(t)$). Therefore, this representation captures the keyword at the beginning and the end alongside its length.

$$\vec{g}_t^{(ep)} = [\vec{h}_{begin(t)}, \vec{h}_{end(t)}, \phi(t)] \#1$$

Self-attention representation is the weighted summation of contextual word vectors in edu_t, as shown in Eq. 2. Each word is assigned a weight which is computed from $\overrightarrow{h_k}$. First, each contextual word vector ($\overrightarrow{h_k}$) is fed to a linear function (f_{gate}). Second, the output vector is applied with Softmax function to calculate the weight for each word.

$$\vec{g}_t^{(atten)} = \sum_{k=begin(t)}^{end(t)} \text{softmax}\left(f_{gate}\left(\overrightarrow{h_k}\right)\right) \cdot \overrightarrow{h_k} \#2$$

4.3 EDU combination scoring

This module is responsible for calculating the scores $s_t \in \mathbb{R}^{\{t-1\}}$ for edu_t where $s_{t,j}$ is a score between edu_t and edu_j. The two proposed methods are inspired by dependency parser and coreference resolution tasks.

Dependency-parsing-based method is based on biaffine dependency parsing, proposed by Dozat and Manning (2016). First, each EDU representation $\overrightarrow{g_t}$ is embedded into the child vector $\vec{h}_t^{(ch)}$ and the parent vector $\vec{h}_t^{(pa)}$ by linear functions (f_{ch} and f_{pa}), as shown in Eq. 3 and Eq. 4.

$$\vec{h}_t^{(ch)} = f_{ch}(\overrightarrow{g_t}) \#3$$

$$\vec{h}_t^{(pa)} = f_{pa}(\overrightarrow{g_t}) \#4$$

After that, both sequences of vectors (H_{ch} and H_{pc}) are applied to bilinear matrix attention to compute the scores $\overrightarrow{s_t}$, which are computed as shown in Eq. 5.

$$\overrightarrow{s_t} = \left(H^{(ch)} \oplus \vec{1}\right) \times U \times \vec{h}_t^{(pa)} \#5$$

Coreference-based method is based on end-to-end coreference resolution, Lee et al. (2017), as shown in Eq. 6. The score of each pair of EDU (edu_t, edu_i) is composed of three parts: two individual scores ($s_t^{(in)}, s_i^{(in)}$) of EDUs and the antecedent score calculated from both EDUs ($s_{t,i}^{(a)}$).

$$s_{t,i} = s_t^{(in)} + s_i^{(in)} + s_{t,i}^{(a)} \#6$$

Both types of scores are calculated based on linear functions (f_{in}, f_a), which are describes below:

$$s_t^{(in)} = w_{in} \cdot f_{in}(\overrightarrow{g_t}) \#7$$
$$s_{t,i}^{(a)} = w_a \cdot f_a\left(\overrightarrow{g_t}, \overrightarrow{g_i}, \overrightarrow{g_t} \circ \overrightarrow{g_i}, \zeta(t,i)\right) \#8$$

where $\cdot$ denotes the dot product, $\circ$ denotes element-wise multiplication, w_{in} and w_a are weights for calculating an individual score and an antecedent score respectively, and $\zeta(t,i)$ is the distance between edu_t and edu_i.

4.4 Inference and training process

This process is fed with combination scores of each EDU. In this case, zero is added to the list of scores to represent an individual EDU. Then, Softmax function is applied to the scores to find the probability distribution of combining edu_t.

The probability indicates if the EDU should be combined and which EDU is combined with. The answer is the position of highest probability y_t, as shown in Eq. 9. If y_t is the index of added zero, the considered EDU is individual. In the other hand, if y_t is other position, the considered EDU is combined with the EDU at y_t.

$$y_t = \text{argmax}\left(\text{softmax}([0, \overrightarrow{s_t}])\right) \#9$$

In the training process, the marginal log-likelihood of all correct combined EDU position ($\hat{y}_t$), which is indicated from the label. The calculation of the loss L is shown in Eq. 10.

$$L = log \prod_{t=1}^{M} P(\hat{y}_t) \ \#10$$

5 Experimental setup

5.1 Dataset

In this work, the dataset is collected from social media sources, including conversations and posts. After that, the text is segmented into EDUs by our in-house model and then combined EDUs from mentioned criteria by linguists. The dataset contains 161,515 arbitrary texts, which can be segmented into 847,186 EDUs. There are 59,564 pairs of EDUs that are combined.

In the annotation process, we also specify which relation in the criteria is used to combine. The number of each relation in the data is shown in Table 1.

type	# of relations
attribution	23,949
attribution-n	3,948
elaboration-object-attribute	23,971
same-unit	7,696

Table 1. The frequency of each relation that is used to combine EDUs

In pre-processing, in-house models that are trained by social media data are used (Lertpiya et al., 2018). The text is tokenized into a sequence of words. After that, each word is tagged with part-of-speech.

In the training and inference process, the dataset is randomly split with a ratio 9:1 for training set and testing set, respectively. After that, we split 10% of training set for validation set. Meanwhile, the rest of training set is truly used for training the model.

5.2 Implementation details

The hyperparameters of the trained model and the optimizer are described in this section. The word and POS embedding sizes are 300 and 100. The word embedding is pre-trained on social media data with the Skip-gram technique (Mikolov et al., 2013). Four layers of Bi-LSTM are stacked. The hidden size of each Bi-LSTM is 32, and there are dropout layers whose rate is 0.1 between the layers. In end-point representation, the size of length embedding $\phi(t)$ is 20.

In the dependency-based scoring method, the hidden sizes of fully connected layers f_{ch} and f_{pa} are both 64. Meanwhile, in the coreference-based scoring method, the embedding size of the distance $\zeta(t, i)$ is 20. The hidden size of fully connected layers f_{in} and f_a are 30 and 150, respectively.

The optimizer is ADAM (Zhang, 2018), whose initial learning rate is 0.001. The learning rate is reduced by 0.5 when the F1 score has stopped improving for five epochs. The batch size is 16. The model is trained 40 epochs and selects the model, which gains the highest F1 score on validation set.

5.3 Evaluation

The metric for this paper should have three characteristics. First, the metric reflects the intersection between combined EDUs label and prediction to measure the performance of the model. Second, since a combined EDUs is not always limited to be constructed from only two EDUs, so the metric needs to consider combining more than two EDUs. Finally, the metric ignores the order or direction of relations that are used to combine because combining EDUs does not need to be interested in the relations.

According to, a coreference resolution task also considers those characteristics for evaluation. Therefore, the metrics from coreference resolution are chosen for evaluation for this paper. However, there are many methods proposed for calculating F1 on coreference resolution. Each method has different advantages and drawbacks. Thus, the average of F1 that used in this paper are calculated from three methods: MUC (Vilain et al., 1995), B^3 (Bagga & Baldwin, 1998), and $CEAF_{\varphi_4}$ (Luo, 2005) as same as Lee et al. (2017).

6 Results

The results of two experiments are discussed in this section. The first experiment compares the different EDU combination scoring method. Meanwhile, the second experiment shows the performance of each EDU representation constructed from contextual word vectors.

6.1 EDU combination scoring

The different of EDU combination scoring methods are compared in the experiment. In this case, EDU representation module is composed of only end-point module. Table 2 shows that scoring with the coreference-based method outperforms the dependency-based method in terms of F1 score. The result occurs because the coreference-based method includes the distance between considered EDUs in a score calculation. Meanwhile, the dependency-based method exploits only the representation of the considered EDUs.

representation	precision (%)	recall (%)	F1 (%)
endpoint	82.81	82.41	82.61
self-attention	83.10	81.82	82.46
both	83.04	82.41	82.72

Table 2. The comparison of each combining scoring method

Figure 7 shows the frequency of the distance between combined EDUs to prove the aforementioned statement. The result indicates that the distance of combined EDUs is usually short.

In other words, if EDUs are far from each other, those EDUs should not be combined. Therefore, the distance is an important feature for the model on combining EDUs. As a result, the coreference-based method, which includes the distance feature, can perform better than the dependency-based method.
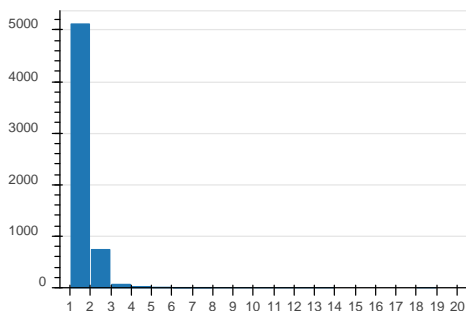

Figure 7. Distance between combined EDU

6.2 EDU representation

In this section, the difference between an endpoint representation and self-attention representation is shown. Table 3 shows the results that, by using end-point representation, the F1 is slightly higher than using self-attention representation.

scoring method	precision (%)	recall (%)	F1 (%)
coreference-based	82.81	82.41	82.61
dependency-based	77.28	83.08	80.06

Table 3. The comparison of each representation

The reason is that most of combined EDUs contain a complementizer ('ว่า', 'ที่') as a marker, which is usually the first word of preceded EDU in the combined EDUs. The statement can be proved in Figure 8. This figure shows that the two most frequent words at the beginning of the preceded EDU in combined EDU are the mentioned complementizers. Therefore, the end-point representation can be trained easier because the representation focuses on the beginning and end words of the EDU.

Instead of using each representation separately, both representations are trained in the model. The model achieves the highest score at 82.72% in terms of F1.
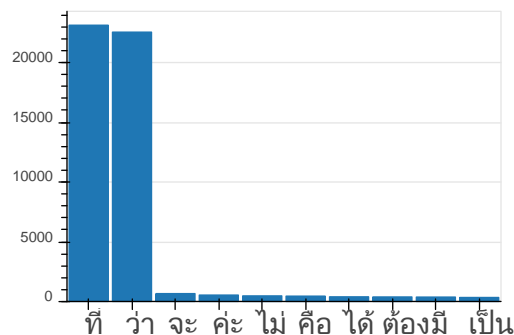

Figure 8. The frequency of the first word on the preceded EDU in the combined EDUs

7 Discussion

According to the results of the above, the discussion of the model is presented in this section. The details are focused on the process that the best model works on each relation for combining EDUs. Table 4 shows the recall of each relation. In this case, because the model does not concern about which class is predicted, so the precision of each relation is not evaluated.

Relation type	Recall (%)
attribution	82.17
attribution-n	95.02
elaboration-object-attribute	71.35
same-unit	53.01

Table 4. Recall score on each relation type that is used to combine EDUs

The model achieves 82.17% and 95.02% on 'attribution' and 'attribution-n'. According to, there is a marker like a complementizer 'ว่า' to indicate that they should be combined.

Meanwhile, 'elaboration-object attribute' is usually indicated with a complementizer 'ที่' as a marker. However, this complementizer is also included as a marker in other RST relations. Therefore, the model achieves only 71.35% recalls on this relation due to the various usage of the marker.

Besides that, the model cannot perform well on 'same-unit' relation, of which recall is only 53.01%. The reason is that there is no marker to classify this relation. Moreover, this relation is rarely found on our dataset. Therefore, there is insufficient information for the model to learn this relation without any marker.

8 Conclusion

In this paper, we propose the criteria of both the syntactic and semantic way for selecting rhetorical relations, which are used to combine EDUs. Moreover, the dataset was annotated by using those criteria before being used to train a deep learning model.

Two experiments were conducted to find the best configuration of the model. In the EDU combination scoring, the coreference-based method achieved a better F1 score. We suspect this because a distance between EDUs is correlated to its likeliness to combine. Meanwhile, using both proposed EDU representations lead to the best F1 score for combining EDUs. The best model achieves 82.72% in terms of F1 score. In an ablation study, the model works well on 'attribution' and 'attribution-n' relation due to the keywords at the beginning of EDU. However, 'same-unit' relation is hard for the model to predict as a relation for combining EDUs because there is no keyword to guide the model and lack of annotated data.

In this paper, we have primarily focused on the combining EDUs task. However, further experiments should be performed to evaluate how the use of combined EDUs affects downstream tasks (e.g., intention classification and sentiment analysis).

References

Bagga, A., & Baldwin, B. (1998). Entity-Based Cross-Document Coreferencing Using the Vector Space Model. 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1, 79–85.

Carlson, L., & Marcu, D. (2001). Discourse tagging reference manual. University of Southern California Information Sciences Institute.

Carlson, L., Marcu, D., & Okurowski, M. E. (2001). Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. Proceedings of the Second SIGdial Workshop on Discourse and Dialogue, 16, 1–10.

Dozat, T., & Manning, C. D. (2016). Deep biaffine attention for neural dependency parsing. arXiv preprint arXiv:1611.01734.

Intasaw, N., & Aroonmanakun, W. (2013). Basic principles for segmenting Thai EDUs. Proceedings of the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 27), 491–498.

Ketui, N., Theeramunkong, T., & Onsuwan, C. (2012). A rule-based method for thai elementary discourse unit segmentation (ted-seg). 2012 Seventh International Conference on Knowledge, Information and Creativity Support Systems, 195–202.

Ketui, N., Theeramunkong, T., & Onsuwan, C. (2015). An EDU-Based Approach for Thai Multi-Document Summarization and Its Application. ACM Trans. Asian Low-Resour. Lang. Inf. Process., 14(1), Article 4.

Lee, K., He, L., Lewis, M., & Zettlemoyer, L. (2017). End-to-end neural coreference resolution. arXiv preprint arXiv:1707.07045.

Lertpiya, A., Chaiwachirasak, T., Maharattanamalai, N., Lapjaturapit, T., Chalothorn, T., Tirasaroj, N.,

& Chuangsuwanich, E. (2018). A preliminary study on fundamental thai nlp tasks for user-generated web content. 2018 International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP), 1–8.

Luo, X. (2005). On coreference resolution performance metrics. Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, 25–32.

Mann, W. C., & Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. Text, 8(3), 243–281.

Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. Proceedings of the ACL Interactive Poster and Demonstration Sessions, 170–173.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems, 26, 3111–3119.

Morey, M., Muller, P., & Asher, N. (2018). A dependency perspective on rst discourse parsing and evaluation. Computational Linguistics, 44(2), 197–235.

Raiman, J., & Miller, J. (2017). Globally normalized reader. arXiv preprint arXiv:1709.02828.

Schauer, H. (2000). From elementary discourse units to complex ones. 1st SIGdial Workshop on Discourse and Dialogue, 46–55.

Singkul, S., Khampingyot, B., Maharattamalai, N., Taerungruang, S., & Chalothorn, T. (2020). Parsing Thai Social Data: A New Challenge for Thai NLP. 2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP), 1–7.

Sinthupoun, S., & Sornil, O. (2010). Thai rhetorical structure analysis. International Journal of Computer Science and Information Security (IJCSIS), 7(1), 95-105.

Sukvaree, T., Charoensuk, J., Wattanamethanont, M., & Kultrakul, A. (2004). RST based Text Summarization with Ontology Driven in Agriculture Domain. Department of Computer Engineering, Kasetsart University, Bangkok, Thailand.

Taboada, M., & Mann, W. C. (2006). Rhetorical Structure Theory: looking back and moving ahead. Discourse Studies, 8(3), 423-459.

Van Lierde, H., & Chow, T. W. (2019). Query-oriented text summarization based on hypergraph transversals. Information Processing & Management, 56(4), 1317-1338.

Vilain, M., Burger, J., Aberdeen, J., Connolly, D., & Hirschman, L. (1995). A model-theoretic coreference scoring scheme. Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995, 45–52.

Zhang, Z. (2018). Improved adam optimizer for deep neural networks. 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), 1–2.