The Journal of Studies in Language

The Journal of Studies in Language 37(1), 037-048 (2021)

ESFWN-based Event Structure Frame Type Classifier for Event Structure-dependent Inferencing

Im, Seohyun
Seoul National University

ABSTRACT

***The Journal of Studies in Language* 37.1, 037-048**. The current study aims to introduce the ESFWN-based Event Structure Frame Type (ESF type) Classifier for English verbs in text. It is a component of the Event Structure-based Inference Generation (ESIG) system we designed to generate event structure-related inferences. The classifier annotates a proper ESF type to a verb in a given sentence using the Event Structure Frame-annotated WordNet (ESFWN) and the Word Sense Disambiguation algorithm named EWISER. The advantage of the classifier is that because its verb classification depends on ESFWN, we only need word sense disambiguation, which maps the target verb to its proper wordnet synset. Given the WordNet synset for the target verb, the classifier annotates the ESF type corresponding to the synset. The F1-score of the classifier is 84.71%. **(Seoul National University)**

Keywords: Event Structure Frame, ESF Type Classifier, Event Structure-based Inference Generation (ESIG) system, ESFWN, EWISER

https://doi.org/10.18627/jslg.37.1.202105.037

pISSN : 1225-4770
eISSN : 2671-6151

Received: April 01, 2021
Revised: April 29, 2021
Accepted: May 17, 2021

1. Introduction

Since Vendler (1967) suggested that verbs have their lexical aspect (Event Structure) and can be classified into one of the four classes – state, process, achievement, and accomplishment, Event Structure has been one of main research topics in verb semantics. Event Structure is a verb meaning representation frame which focuses on the lexical aspect of a verb (Pustejovsky, 1995). For example, the verb *arrive* in (1a) has its event structure as in (1b).

(1) a. *John **arrived** at school at 9 o'clock this morning.*
b. ***arrive*** [move_to_goal]
se1: pre-state: not_be_at (John, school, t1)
se2: process: arriving_at (John, school, t2)
se3: post-state: be_at (John, school, t3)

- This work was supported by Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. 0418-20210010 (section 2) developing technology of learning the event-situation knowledge system and recognizing events and inferencing relations for video understanding)

In the sentence (1a), the meaning of *arrive* is decomposed to the three subevents − pre-state (se1), process (se2), and post-state (se3) as presented in (1b). se1 (pre-state) represents John's state **before** his arriving at school. se2 (process) means the process of John's arriving at school and se3 (post-state) represents John's state **after** his arriving at school. The three subevents are ordered temporally. Semantically, the pre-state is a presupposition of the main event − arriving − and the post-state is its resultative entailment.

From the viewpoint of Natural Language Inference (NLI)[1], a subfield of Natural Language Processing (NLP), the Event Structure-based decompositional meaning representation of a verb in a sentence enables an NLI system to extract inferences based on the subeventual structure of the verb, even though the inferences are not explicitly expressed. Let's see some questions and answers in (2).

(2) Questions and Answers regarding the sentence in (1a)

Q1: Was John at school **before** 9 o'clock this morning?	A1: No.
Q2: Was John at school **right after** 9 o'clock this morning?	A2: Yes.
Q3: Where was John **right after** 9 o'clock this morning?	A3: at school.

We, human, can give right answers to the above questions naturally, although the sentence in (1a) gives no explicit information about John's location before or after his arriving at school. Those inferences are dependent on the event structure of the verb *arrive* in the sentence (1a).

Although most NLI systems with deep learning algorithms and huge corpora are successful in extracting some kinds of inferences from natural language text, they still have difficulty in deriving every kind of inferences human can extract from text. To help resolving the issue, we are developing the <u>E</u>vent <u>S</u>tructure-based <u>I</u>nference <u>G</u>eneration (ESIG) System that automatically extracts inferences based on the event structure of verbs in text. In this paper, we focus on introducing the ESFWN[2]-based Event Structure Frame (ESF) Type Classifier, a component of the ESIG System.

The structure of this paper is as follows. First, we introduce the ESIG System briefly as a big picture in the next section. Section 3 is a key part of this paper that explains what the ESFWN-based ESF Type Classifier is. We describe our experiment with the classifier in section 4 and mention other studies related with the ESF Type Classifier in section 5. In section 6, we conclude and mention some future work.

2. The Event Structure-based Inference Generation (ESIG) System

The ESIG System gets a sentence (text) as its input and print out the event structure-dependent inferences by applying its component algorithms to the input sentence in order. For instance, if we type the sentence (1a) as an input to the ESIG System, then it gives us inferences like the sentences in (3).

1) Natural Language Inference (NLI) is a subfield of Natural Language Processing (NLP) which concentrates on developing the algorithm that extracts or generates inferences from language text.

2) ESFWN is an acronym of the Event Structure Frame-annotated WordNet. We explain what ESFWN is in more detail in section 3.1.

(3) ESIG system's output: inferences from the sentence (1a)
a. John **was not at** school before 9 o'clock this morning.
b. John **was at** school right after 9 o'clock this morning.
c. John **came to** school at 9 o'clock this morning.
d. John **got to** school at 9 o'clock this morning.

The basic idea of the ESIG system is to generate event structure-related inferences from a verb and its arguments in text using language resources (or linguistic knowledge bases) and algorithms, assuming that verbs are classified into one of the pre-defined Event Structure Frame types depending on its context because all verbs in one class share the same Event Structure Frame and has the same kind of event structure-based inferences. In (3), the verbs such as *arrive*, *come*, and *get* are classified into the move_to_goal ESF type and share the ESF presented in (4). Therefore, the verbs all extract the inferences in (3a, b) depending on their event structure.

(4) move_to_goal ESF type and its ESF
se1: pre-state: not_be_at (agent/theme, goal_location, t1)
se2: process: V-ing (agent/theme, goal_location, t2)
se3: post-state: be_at (agent/theme, goal_location, t3)

Figure 1 presents the procedure of generating event structure-related inferences by applying the ESIG system to an input sentence. Given the input sentence, the ESIG system first applies its ESF Type Classifier to the input and determine what is the ESF type of the verb in the input sentence. We defined 69 ESF types previously and listed those together with their corresponding ESFs in the ESF-dictionary. The next step is to call its corresponding ESF from the ESF-dictionary. After the system gets the ESF, it inserts the appropriate text to the argument and predicate positions of the ESF by the Argument Structure Annotation and Predicate Insertion algorithm. The last step of the ESIG system is to change each subevent in the ESF to a sentence by applying the Inference Sentence Generator. The final result of the procedure is event structure-related inference sentences generated from the given sentence.

Under the present condition of Natural Language Inference (NLI) field, it is necessary to develop a system for an Event Structure-dependent inference generation or recognition such as the ESIG system, since deep learning-based NLI systems still are not successful in inferencing various kinds of inferences human can recognize or generate including event structure-dependent inferences (Kober et al., 2019). Moreover, the ESIG system can be used to extend the existing NLI corpus like Stanford Natural Language Inference dataset (SNLI dataset; Bowman et al., 2015) by adding the resulting data annotated by the ESIG system to the SNLI dataset.

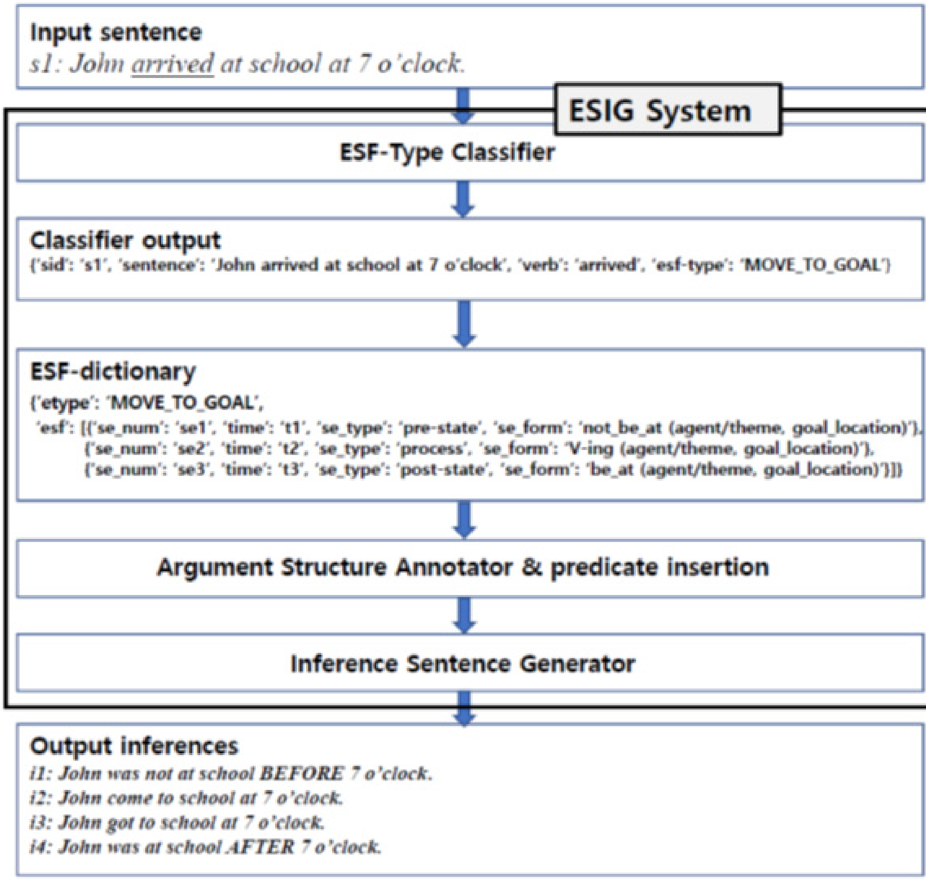

Fig. 1. 1 Event Structure Frame-based Inference Generation (ESIG) System

3. ESFWN-based Event Structure Frame Type Classifier

In section 2, we described the whole system of Event Structure-based Inference Generation (ESIG) we are developing for NLI tasks. Now, we introduce the ESFWN-based Event Structure Frame Type Classifier (ESF Type Classifier), a main component of the ESIG system. In section 3.1, we first describe the ESF-dictionary and ESFWN used as resources for the classifier. Section 3.2 introduces the procedure of the ESFWN-based ESF Type Classifier.

3.1 ESF-dictionary and ESFWN

The basic idea of Event Structure Frame (ESF) is originated from Im (2013) and Im and Pustejovsky (2009, 2010). ESF is a framework for generalizing the common subeventual structure of events represented by a class of verbs. For example, the verbs *run* and *walk* have the ESF for move type in common, because they belong to the verb class move. Im (2013) proposes 23 pre-defined ESF types and their corresponding ESFs. We extend the ESF types to 69 types as shown in (5) so that our ESIG system can generate as many inferences as possible from text by using the ESF types (Im, 2018, 2019, 2021). We built the ESF-dictionary which has 69 pairs of ESF type and its corresponding ESF. You can see how a pair <ESF type, ESF> is composed in Figure 1.

(5) ESF types in the ESF-dictionary[3)]

process (cause-), semelfactive, maintain, move (cause-), move_from_source (cause-), move_to_goal (cause-), move_from_source_to_goal (cause-), move_forward (cause-), move_back (cause-), move_up (cause-), move_down (cause-), move_toward_speaker (cause-), move_around (cause-), pass (cause-), accompany, carry, spread (cause-), lose (cause-), get (cause-), give, take_cop, exchange, get_info (cause-), info_transfer, change_state (cause-), become (cause-), change_direction (cause-), begin (cause-), continue (cause-), end (cause-), positive_causation, negative_causation, happen, precede, follow, performative.

A critical innovation of our ESFWN-based ESF Type Classifier is to annotate its proper ESF type to each synset of verbs in WordNet (Miller, 1995), which has more than 24,000 synsets for over 2,400 English verbs. We annotated a proper ESF to each synset to build the Event Structure Frame-annotated WordNet (ESFWN; Im, 2018, 2019). For instance, the English verb *kill* has 15 synsets in WordNet. Each of them has its own ESF type annotated in ESFWN. We show some synsets in (6) to explain how the ESF type is annotated in the ESFWN.

(6) *Kill* in WordNet

a. **kill.v.01** (cause to die; put to death, usually intentionally or knowingly)
"This man killed several people when he tried to rob a bank"; "The farmer killed a pig for the holidays."

b. **kill.v.14** (cause to cease operating) "kill the engine"

As you see in (6), a WordNet synset entry includes its synset number, definition, and examples. Interestingly, kill.v.01 and kill.v.14 have different ESF types: kill.v.01 synset has cause_go_out_of_existence ESF type but the other has cause_end type. As a result, the inferences derived from their ESFs are totally different. In the ESFWN, the ESF types for the synsets of kill.v.01 and kill.v.14 are annotated as follows:

(7) ESF type annotation example in the ESFWN

a. {"VERB": "kill", "SENSE_NUMBER": "kill.v.01", "SENSE_KEY": "kill%2:35:00::",
"OFFSET": "wn:01323958v", **"ESF_TYPE": "CAUSE_GO_OUT_OF_EXISTENCE"**,
"SYNONYMS": ["kill"], "HYPERNYMS": [], "VID": 12024},

b. {"VERB": "kill", "SENSE_NUMBER": "kill.v.14", "SENSE_KEY": "kill%2:30:03::",
"OFFSET": "wn:00355365v", **"ESF_TYPE": "CAUSE_END"**, "SYNONYMS": ["kill"],
"HYPERNYMS": ["switch_off", "cut", "turn_off", "turn_out"], "VID": 12035}

As you see in (7), each entry of ESFWN is a python dictionary that has keys like verb_lemma, sense_number (synset number), sense_key, offset, esf_type, synonyms, and hypernyms. ESFWN has total 24601 entries for around 2,400 English verbs[4)]. Please read Im (2018, 2019, 2021) to get more specific information about ESFWN and how we built it.

3) (cause-) means the causative counterpart of the ESF type. For example, move (cause-) refers to move and cause_move ESF types. The ESF-dictionary can be found at https://github.com/ish97/ESFWN-based-ESL-Annotator/blob/main/script_esf/esf_lib.py

4) ESFWN is at github.com/ish97/ESFWN-based-ESL-Annotator/blob/main/script_esf/esfwn_v1.json.

3.2 ESFWN-based Event Structure Frame Type Classifier

In the previous section, we described the ESF-dictionary and the ESFWN which support the ESFWN-based ESF Type Classifier. Now, we explain the classifier in more detail. Given a sentence, the classifier outputs the correct ESF types for the verbs in the sentence via the ESF type classification procedure we show in Figure 2.

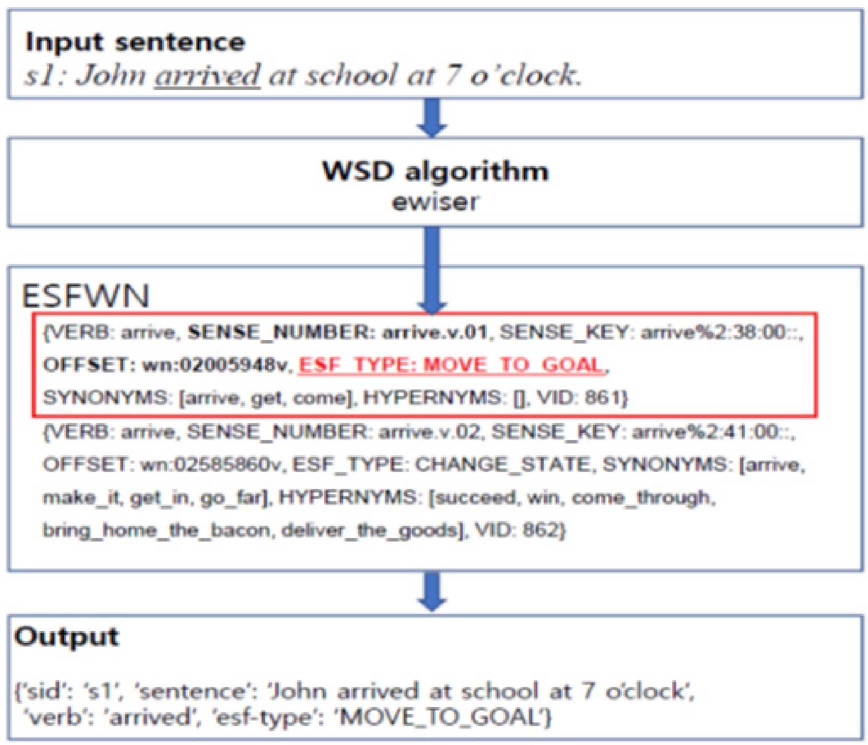

Fig. 2. Classification procedure of the ESF Type Classifier

Since the ESF type for each synset is already determined, the performance of the ESF Type Classifier depends entirely on a Word Sense Disambiguation (WSD) algorithm which maps a target verb in each sentence to its proper synset in ESFWN. Figure 2 shows the procedure of annotating its ESF type to the verb *arrive* when the ESFWN-based ESF Type Classifier is applied to the input sentence. Given a sentence as in Fgure 2, the ESF type classifier calls a WSD algorithm named EWISER (Bevilacqua and Navigli, 2020). The result of applying EWISER to the sentence is in (8).

(8) EWISER WSD result for the sentence *John arrived at school 7 o'clock in the morning*.
"John@#*John@#*PROPN@#* **arrived**@#*arrive@#*VERB@#***wn:02005948v**
at@#*at@#*ADP@#* school@#*school@#*NOUN@#*wn:08276720n
7@#*7@#*NUM@#* o'clock@#*o'clock@#*NOUN@#* in@#*in@#*ADP@#*
the@#*the@#*DET@#* morning@#*morning@#*NOUN@#*wn:15165289n
.@#*.@#*PUNCT@#* \n", '\n'

Given a sentence, EWISER disambiguates the meaning of each word contextually and annotates an appropriate WordNet synset offset to the word, if possible. As presented in (8), EWISER prints <token, lemma, part-of-speech, WordNet offset number> with the delimiter @#* as its output. For example, **arrived**@#*arrive@#*VERB@#***wn:02005948v** means that a token is *arrived*, its lemma is *arrive*, it is a verb and its WordNet offset is wn:02005948v.

We need only WordNet offset of a verb out of the result of applying EWISER to the input sentence, because the ESF Type Classifier selects one of the entries in the ESFWN using the offset as a key. In (8), the offset of *arrive* is 'wn:02005948' which corresponds to the synset number 'arrive.v.01' in ESFWN. The next step is to find the entry which corresponds to the synset in ESFWN. Finally, we get the ESF type of *arrive*, move_to_goal, by getting the value of the key 'ESF_TYPE' in the entry of ESFWN.

EWISER is the acronym of the Enhanced WSD Integrating Synset Embeddings and Relations. According to the authors, it first broke through the 80% ceiling on the concatenation of all the standard all-words English WSD evaluation benchmarks. EWISER achieved great performance of WSD by embedding information from the Lexical Knowledge Base (LKB) graph within the neural architecture for WSD and exploiting pre-trained synset embeddings, which makes it possible for the network to predict synsets even though those are not in the training set. The code and pretrained models are at github.com/SapienzaNLP/ewiser.

For verb classification, we can use deep learning algorithms and huge data, which have recently been in the spotlight, or traditional methods with language resources or knowledge bases . In this research, we adopt a method that exploits a lexical resource called ESFWN rather than a large dataset and a deep learning algorithm. The advantage of this method over deep learning approaches is that it does not cost to build huge datasets because it uses only ESFWN and a WSD algorithm. Second, we can improve the performance of the ESF Type Classifier only by upgrading the WSD algorithm. The current SoTA in WSD is EWISER but we can embed better WSD algorithm to the ESF Type Classifier at any time. Third, the usefulness of ESFWN is that it makes it possible to use all kinds of lexical semantic information WordNet provides as well as event structure of a verb, since the ESFWN is linked to WordNet.

4. Experiment

The ESFWN-based ESF Type Classifier aims to print the ESF type of a target verb as its output when an English sentence is given as its input. We did an experiment with a small dataset collected from the Stanford Natural Language Inference corpus to test the ESF type classifier. In this section, we present the experiment result.

4.1 Data

The SNLI corpus is a huge corpus which includes pairs of <sentence, inference>. We chose 163 sentences with change_of_location verbs out of the sentences in the SNLI corpus. To build a Gold-Standard dataset, a linguist annotated a proper ESF type to only a target verb in each sentence, even when the sentence has more than one verb.[5] All verb lemmas are listed below:

5) It means a sentence can be included in our test dataset several times, with different target verbs.

(9) change_of_location verb lemmas used for test (total 60 lemmas)
advance, airlift, appear, bring, carry, chase, climb, cross, dance, descend, direct, drop, emerge, exit, fall, fetch, get, go, hang, hit, jog, jump, kick, lead, leap, lift, load, make, march, paddle, pass, place, pop, pour, pull, push, put, race, raise, rappel, reach, ride, roll, run, sink, skate, sled, slide, splash, stack, stoop, stop, stroll, surf, swim, throw, tread, walk, wave, zip

Depending on the sense (synset) of a verb, its contextual ESF types can be different. For instance, the verb *cross* has different ESF types in the two sentences of (10).

(10) ESF type of the verb *cross* in context
a. *A man sitting on a chair with his legs* **crossed** *looking off to the right.*
[cross.v.04; cause_change_posture]
b. *A man and a woman* **cross** *the street in front of a pizza and gyro restaurant.*
[traverse.v.01; pass]

4.2 Experiment Results

We applied the ESFWN-based ESF Type Classifier to total 163 verbs in our test data. As a result, we got the ESF type annotations for total 151 verbs and failed to annotate the target verbs in 12 sentences. The failure of annotation for the 12 examples is caused by EWISER or EWISER-wrapper.[6] It includes part-of-speech recognition errors. If EWISER annotates the part-of-speech of a target verb in our test data wrongly, the EWISER-wrapper fails to catch the target verb because the wrapper is designed to collect only information regarding verbs. Out of total 151 verbs annotated, 133 verbs have correct ESF types. The wrong annotations include the WSD errors caused by EWISER and ESF type errors.

We show some examples with wrong ESF type annotation in (11). The verb *directed* is a target verb in the first example sentence of (11a). Its synset is annotated as lead.v.01 by EWISER. Depending on its synset, the ESF Type Classifier annotates its ESF type as accompany. However, it should be interpreted as the sense of direct.v.09 and thus its ESF type must be process. *gotten* in (11b) is annotated wrongly as become type but belongs to move_to_goal class. The verb *advance* in (11c) is move_forward type but the ESF type classifier annotates it as get type, for EWISER gave a wrong synset for the verb in the sentence.

(11) examples with wrong annotations
a. *People wait patiently in the crowd by the firetruck as they are* **directed** *by the fireman.*
(lead.v.01, accompany; direct.v.09, process)
b. *A small boy has* **gotten** *into the cabinet and gotten flour and crisco all over himself.*
(become.v.01, become; arrive.v.01, move_to_goal)
c. *A soccer game where the team in yellow is attempting to* **advance** *past the team in white....*
(gain.v.05, get; advance.v.01, move_forward)

6) The EWISER-wrapper calls EWISER and get the results we need from the output of EWISER application.

When we include the verbs which the classifier failed to annotate as error data, test data are composed 151 verbs and Gold-Standard has 163 verbs. Then, precision is 88.08% and recall is 81.60%. F1-score is 84.71%. On the other hand, if we exclude the unannotated data (12 verbs), precision, recall, and F1-score all are 88.08%. Table 1 summarizes the ESF Type Classifier performance. If we assume our ESFWN has a proper ESF type for each synset of verbs in WordNet, F1-score at least over 80% is natural because the WSD algorithm, EWISER, has higher F1-score than 80% in its WSD evaluation.

Table 1. ESF type Classification Evaluation

	precision	recall	F1-score
Including unannotated data	88.08%	81.60	**84.71%**
Excluding unannotated data	88.08%	88.08%	88.08%

5. Related Work

ESFWN and ESF Type Classifier are components of the Event Structure-based Inference Generation (ESIG) system, whose goal is to get implicit information related to events denoted by verbs in text. There have been a few linguistic resources and automatic event structure annotation tools to achieve the goal.

First of all, the pre-defined ESF types used for the ESFWN-based ESF Type Classifier depends on GESL − the Generator of the Event Structure Lexicon - developed by Im (2013) and Im and Pustejovsky (2009, 2010). GESL annotates automatically event structure frame and other lexical and grammatical information such as tense, aspect, synonyms, and hypernyms to each verb in a sentence. The main components of GESL are the ESF type dictionary and the ESF type classifier which uses a machine learning algorithm - Support Vector Machine[7]. We extend the ESF types suggested by GESL to 69 types by subdividing the ESF types in GESL and adding new types. Its classification performance is F-measure 77.48%. Even though we cannot compare it directly with our ESFWN-based ESF type classifier, the performance of our classifier is around 7% higher than that of GESL.

Segers et al. (2015, 2016) made an ontology named the Event and Situation Ontology (ESO) for similar purpose as ours, that is, to extract implicit information about the sub-eventual structure of an event denoted by a verb in text. The ESO is a resource which formalizes the pre and post conditions of events and the roles of the entities affected by an event. The ESO is integrated into the Predicate Matrix, an automatic extension of SemLink (Palmer, 2009) that merges several models of predicates such as VerbNet, FrameNet, PropBank and WordNet. They tested the ESO with the NewsReader pipeline and reported a relatively low precision and recall but argued that it showed promising results with respect to a quality.

On the other hand, FrameNet based on Frame Semantics (Fillmore, 1976) has no structural information about the subevent semantics of English verbs, although it has more detailed semantic role information of verbs. VerbNet (Kipper

7) Support Vector Machine (SVM) is a learning model used in machine learning prior to deep learning. Given a training dataset, SVM generates a non-probable linear classification model that determines which category the new data will fall into based on that dataset. It is one of the best and most popular machine learning models.

2005) also has difficulty in using it directly to extract information about verbs' event structure, if we use the original VerbNet frames (Zaenen et al., 2008). Brown et al. (2019) recently changed the frames of VerbNet to reflect the subevent semantics of verbs and made an automatic subeventual structure annotation tool. However, because it still uses its own classes it does not cover all verbs in WordNet.

Next, we mention Kalm et al. (2019) and Kober et al. (2019). Kalm et al. (2019) proposes causal networks for verb meanings to represent event structure and argument structure. An interesting point of this approach is to define the argument structure of a verb in text by a small set of force dynamic relations. Kober et al. (2019)'s work is about grammatical variation of subeventual structure of verbs caused by tense and aspect rather than lexical semantic properties of verbs. For example, the progressive aspect form of a change-of-location verb does not entail the result state − mover's being at goal-location. They point out the deep learning-based NLI systems cannot recognize the entailment cancellation triggered by the aspectual property of a verb in text. Moreover, they argue that simple ruled-based system gives better result than deep-learning based NLI system in recognizing the event structure-dependent inferences. We need to reflect the contextual variation of event structure-based inferences such as cancellation of subevents caused by tense and aspect.

We introduced some lexical resources or ontologies that represent event structure or argument structure of English verbs. It does not seem to be easy, for now, to compare their performance with that of the ESFWN-based ESF Type Classifier directly in this paper, since they have no automatic event structure annotation algorithm (or system) that can be applied to the same data as those we used for now. However, we plan to compare the event structure-related inferencing system or resources by making some criteria for evaluating each of them in the near future.

6. Conclusion and Future Work

Natural Language Inference (NLI) has recently emerged as one of the most interesting and important areas of Natural Language Processing (NLP), along with Natural Language Understanding (NLU). NLI systems based on deep learning and huge inference datasets show good performance, but they do not yet produce inferences as diverse as human give. In particular, those systems cannot recognize or generate inferences related to event structure of verbs in text yet.

In this paper, we propose the ESFWN-based ESF Type Classifier − a main component of the Event Structure-based Inference Generation (ESIG) system we are developing to generate some special kind of inferential sentences based on event structure of verbs from a given sentence. The ESFWN-based ESF Type Classifier is an automatic event structure frame type annotation algorithm which only uses the Event Structure Frame-annotated WordNet (ESFWN; Im, 2021) and the state-of-the-art Word Sense Disambiguation algorithm called EWISER (Bevilacqua and Navigli. 2020). The ESFWN is a lexical resource in which each WordNet synset of English verbs has its proper event structure frame type. EWISER is the best-performing WSD algorithm which embeds information from the Lexical Knowledge Base (LKB) graph within the neural architecture for WSD and exploits pre-trained synset embeddings.

Given the input sentence, the ESFWN-based ESF Type Classifier first calls EWISER to recognize verbs in the sentence and annotate WordNet synset offsets to the verbs. The second step is to find the corresponding entry to each verb in the ESFWN using the WordNet synset offset as a key. Finally the classifier prints out the ESF type of the verb. We got about 84% F-score in a test experiment of the ESF Type Classifier.

The ESF type classifier does not cost to build mass datasets because it uses only ESFWN and a WSD algorithm. Moreover, it is possible to improve the performance of the ESF Type Classifier by replacing the WSD algorithm with a better WSD algorithm. ESFWN is very useful since it is linked to WordNet.

After getting the ESF type of a verb with the ESF Type Classifier, we need to take the event structure frame for the ESF type and insert specific texts into argument positions in the frame. For that purpose, we will develop an argument structure annotator. The last step in the ESIG system will be to generate inference sentences by changing each subevent to a sentence.

As mentioned above, data-driven Natural Language Inference systems with deep learning do not succeed in generating event structure-based inferences. We hope the ESIG system can fill in the gap. One way of doing it is to make a dataset for event structure-based inference. Another way is to add ESFWN as a knowledge base to machine learning-based NLI algorithms. More interestingly, we can apply the ESIG system to infer pre- and post-state of an event or an activity in video. Multimodal extraction of implicit information of a sentence, by combining the ESIG system and visual object recognition, also will be interesting research topic.

References

Bevilacqua, M. and R. Navigli. 2020. Breaking Through the 80% Glass Ceiling: Raising the State of the Art in Word Sense Disambiguation by Incorporating Knowledge Graph Information. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics,* 2854-2864. ACL. https://www.aclweb.org/anthology/2020.acl-main.225.

Bowman, S. R., G. Angeli, C. Potts and C. D. Manning. 2015. A Language Annotated Corpus for Learning Natural Language Inference. *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 632-642, Lisbon, Portugal. Association for Computational Linguistics.

Brown, S. W., J. Bonn, J. Gung, A. Zaenen, J. Pustejovsky and M. Palmer. 2019. VerbNet Representations: Subevent Semantics for Transfer Verbs. *In Proceedings of the First International Workshop on Designing Meaning Representations*, 154-163, Florence, Italy, August 1st, 2019. Associations for Computational Linguistics.

Fillmore, C. J. 1976. Frame Semantics and the Nature of Language. *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech* 280, 20-32.

Im, S. 2021. ESFWN: Event Structure Frame-annotated WordNet for Natural Language Inference Tasks, *Eoneohak* 89, 173-195. The Korean Society of Linguistics.

Im, S. 2019. Semi-Automatic Annotation of Event Structure, Argument Structure, and Opposition Structure to WordNet by Using Event Structure Frame. *In Proceedings of the 10th Global WordNet Conference*, 2019. 07, 23-27, Poland. GWC 2019.

Im, S. 2018. Semi-automatic Annotation of Event Structure Frame to WordNet Synsets. *In Proceedings of the 30th Annual Conference on Human and Cognitive Language Technology,* 101-105.

Im, S. 2013. The Generator of the Event Structure Lexicon (GESL): Automatic Annotation of Event Structure for Textual Inference Tasks. *PhD Dissertation*. Brandeis University, MA, USA.

Im, S. and J. Pustejovsky. 2010. Annotating Lexically Entailed Subevents for Textual Inference Tasks. *In Proceedings of FLAIRS 23*, Daytona Beach, Florida, USA, 2010.

Im, S. and J. Pustejovsky. 2009. Annotating Event Implicatures for Textual Inference Tasks. *In Proceedings of the 5th International Conference on Generative Approaches to the Lexicon*, Pisa, Italy, 2009.

Kalm, P., M. Regan and W. Croft. 2019. Event Structure Representation: Between Verbs and Argument Structure Constructions. *In Proceedings of the First International Workshop on Designing Meaning Representations*, 100-109, Florence, Italy, August 1st, 2019. 2019 Association for Computational Linguistics.

Kipper, K. S. 2005. VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon. *PhD Dissertation*. University of Pennsilvenia.

Kober, T., S. Vroe and M. Steedman. 2019. Temporal and Aspectual Entailment. *arXiv preprint arXiv:*1904.01297 (2019).

Miller, G. A. 1995. Wordnet: A Lexical Database for English. *Communications of the ACM* 38-11 (1995).

Palmer, M. 2009. Semlink: Linking Propbank, VerbNet, and FrameNet. *In Proceedings of the Generative Lexicon Conference*, 9-15.

Pustejovsky, J. 1995. *The Generative Lexicon.* The MIT Press.

Segers, R., M. Rospocher, P. Vossen, E. Laparra, G. Rigau and A. Minard. 2016. The Event and Implied Situation Ontology (ESO): Application and Evaluation. *In Proceedings of the Tenth International Conference on Language Resources and Evaluation* (LREC'16).

Segers, R., P. Vossen, M. Rospocher and L. Serafini. 2015. ESO: A Frame-based Ontology for Event and Implied Situations. In *Proceedings of MAPLEX* 2015.

Vendler, Z. 1967. *Linguistics in Philosophy*. Itacha, NY.: Cornell University Press.

Zaenen, A., C. Condoravdi and D. Bobrow. 2008. The Encoding of Lexical Implications in VerbNet. *In Proceedings of LREC* 2008. Morocco, 2008.

Im, Seohyun, Postdoctoral fellow
1, Gwanak-ro, Gwanak-gu, Seoul, 08826, Republic of Korea
Automation and Systems Research Institute (ASRI), Seoul National University
E-mail: seohyunim71@gmail.com