

Survey of Temporal Information Extraction

Chae-Gyun Lim*, Young-Seob Jeong**, and Ho-Jin Choi*

Abstract

Documents contain information that can be used for various applications, such as question answering (QA) system, information retrieval (IR) system, and recommendation system. To use the information, it is necessary to develop a method of extracting such information from the documents written in a form of natural language. There are several kinds of the information (e.g., temporal information, spatial information, semantic role information), where different kinds of information will be extracted with different methods. In this paper, the existing studies about the methods of extracting the temporal information are reported and several related issues are discussed. The issues are about the task boundary of the temporal information extraction, the history of the annotation languages and shared tasks, the research issues, the applications using the temporal information, and evaluation metrics. Although the history of the tasks of temporal information extraction is not long, there have been many studies that tried various methods. This paper gives which approach is known to be the better way of extracting a particular part of the temporal information, and also provides a future research direction.

Keywords

Annotation Language, Temporal Information, Temporal Information Extraction

1. Introduction

Documents are used to deliver information to the readers. In the past, readers were human, but computers are becoming a new class of readers. Computers can collect information much faster than humans can, and are capable of storing much more information than humans are. To realize these strengths of computers, it is necessary to develop techniques for extracting information from documents. This is because such documents are usually unstructured text. The techniques can be thought of as converters that take unstructured texts as input and output the information in a particular format more favorable for computers. Due to the exponentially increasing number of unstructured documents available on the web and from other sources, developing such techniques is becoming more important.

Among the many aspects of extracting information from documents, the extraction of temporal information has recently drawn much attention. This is because documents usually contain temporal information that is useful for further application in such as knowledge base (KB) construction, information retrieval (IR) systems, and question answering (QA) systems. Given a simple question, “Who was the president of South Korea eight years ago?”, for example, a QA system may have difficulty finding

* This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Manuscript received January 3, 2019; first revision April 9, 2019; accepted April 18, 2019.

Corresponding Author: Young-Seob Jeong (bytecell@sch.ac.kr)

* School of Computing, Korea Advanced Institute of Science and Technology, Daejeon, Korea (rayote,hojinc)@kaist.ac.kr

**Dept. of Big Data Engineering, Soonchunhyang University, Asan, Korea (bytecell@sch.ac.kr)

the right answer without correct temporal information about when the question was posed and what ‘8 years ago’ refers to.

In this paper, studies related to extraction of temporal information are discussed. The relevant studies are summarized in chronological order, and the history of the annotation languages and shared tasks is described. Answers to the following questions are provided in this paper.

- What is temporal information?
- Is there a structured way to describe the task boundary of the temporal information extraction?
- What is the history of the annotation languages?
- What is the history of related studies?
- Are there some shared tasks related to temporal information extraction?
- Which applications can benefit from temporal information?
- What are the research issues?
- How might a system of temporal information extraction be evaluated?

The rest of this paper is organized as follows. Section 2 provides the definition of temporal information, and describes how to represent temporal information. It also gives the definition of the task of temporal information extraction. Section 3 introduces the task boundary of temporal information extraction, and Section 4 gives the history of the annotation languages and shared tasks. Section 5 provides the history of related studies in chronological order, and in Section 6, the research issues are discussed. Some metrics are provided in Section 7, which could be used to evaluate a system of temporal information extraction, along with some issues related to the evaluation process. Finally, Section 8 concludes the paper.

2. Temporal Information

2.1 What Is Temporal Information?

Information can be defined as data endowed with meaning and purpose [1]. Information is inferred from data and is differentiated from data in that it is useful. From a practical point of view, data is a set of raw observations, and information is something useful extracted or inferred from the observations. Information is used to construct knowledge, while wisdom is defined in terms of knowledge. The relationship between these four concepts is depicted in Fig. 1, where the upper concepts are more meaningful and useful than the lower concepts. If information is poorly extracted from data, then it will harm the quality of knowledge and eventually harm the quality of wisdom. Therefore, it is important to develop an effective method for information extraction.

Time can be defined as a measure in which events can be ordered from the past through the present into the future, and also, as the measure of the durations of events and the intervals between them [2]. Based on the definition of information and the definition of time, *temporal information* can be defined as information that can be used to order events and to measure the durations or intervals of events. It is obvious that, to order the events or to measure the durations of the events, it is necessary to take the information about the events into account. That is, the temporal information includes not only the temporal points and durations, but also the information about the events themselves. Furthermore, it is also necessary to consider the relation between the temporal points (or durations) and the events, because such relation has a crucial role for ordering the events or measuring the durations or intervals of the events.

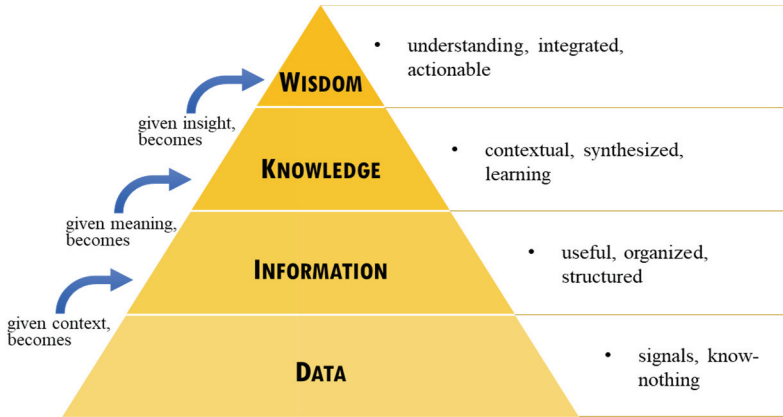


Fig. 1. DIKW (data-information-knowledge-wisdom) pyramid [1].

POS tagging may be useful- also explore if NLP is helpful

To make a concept of temporal information easier to understand, Fig. 2 shows simple examples of them. A green shape with a dashed line means a **time information** (i.e., temporal points or duration), an orange shape with a solid line means an **event**, a double-headed arrow means a **temporal relationship between other entities**, and an underscored word (or phrase) means a **connective which is in the temporal perspective**. At the first example, a verb 'study' and an adverb phrase 'three days' can be event and time information on the given sentence, respectively. A connective 'for' has a role to make a temporal relationship that the 'study' event will be continued during 'three days'. On the other example, there are two events—a verb 'eats' and a noun phrase 'the final exam'—and a temporal relationship between them derived from a connective 'after'. This relationship means that the 'eats' event will be started after another event.

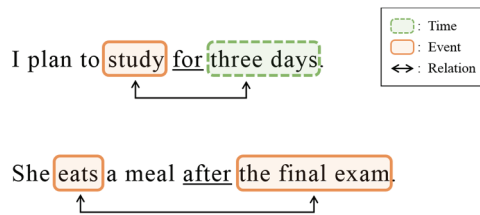


Fig. 2. Examples of temporal information.

Formally, temporal information can be represented as $\{T, E, R\}$, where T denotes the temporal points, durations or intervals, E means the events, and R represents the temporal relation. The relation R can be either of R_{TT} , R_{EE} , or R_{TE} , where R_{TT} denotes the relation between two temporal points (or durations), R_{EE} denotes the relation between two events, and R_{TE} denotes the relation between a temporal point (or duration) and an event. Of course, there can be situations when there is no relation even though there are some T and E .

2.2 Temporal Information Representation

Temporal information appears in raw text through *temporal expression* and *event expression*. Event expression is used to represent the events, while temporal expression is used to denote the temporal

points, durations, and intervals.

[3] suggested that there are three forms of temporal expression: an explicit reference, an indexical reference, and a vague reference. The form of explicit reference directly represents the value of temporal information (e.g., ‘April 4’, ‘March 8, 1983’), while the form of indexical reference indirectly represents the value by relative expressions (e.g., ‘three months later’, ‘yesterday’). The form of vague reference represents ambiguous temporal information (e.g., ‘early 1990s’, ‘about three months’). Meanwhile, [4] suggested that there are three forms of temporal information representation: an explicit reference, an implicit reference, and a relative reference. The explicit reference is same as the explicit reference of [3], while the indexical reference of [3] seems that it contains the implicit reference and relative reference of [4]. The vague reference is absent in the proposed forms of [4].

In this paper, five reference forms are defined: *explicit reference*, *implicit reference*, *relative reference*, *vague reference*, and *non-consuming reference*.

- The form of *explicit reference* directly represents the value of temporal information (e.g., ‘March 8’, ‘2000.08.12’). This form was first mentioned in the fifth Message Understanding Conference (MUC-5) [5].
- The form of *implicit reference* represents a period or a time point that is known by the public without containing any explicit value of temporal information (e.g., ‘the Japanese colonial period’, ‘Middle Ages’). This form can be divided into two subforms, a *global implicit reference* and a *local implicit reference*. The global implicit reference includes temporal expressions that are supposed to be known to the general public, such as ‘Middle Ages’, ‘glacial epoch’, and ‘the Roman Era’. The local implicit reference includes temporal expressions that are supposed to be known to readers. For example, if a document has two sentences “Hoyeon was born 1986.” and “When she was born, the building was established.”, then the readers know the value of the expression ‘When she was born’ which can be inferred by considering the first sentence. The difference between the global implicit reference and the local implicit reference is that the normalized value of the global implicit reference is obtained from a common-sense or external KB, while the normalized value of the local implicit reference is obtained from the information within the corresponding document.
- The form of *relative reference* represents expressions that can be used to infer the value (e.g., ‘two weeks ago’). This form was first mentioned in MUC-7 [5].
- The form of *vague reference* represents ambiguous temporal information (e.g., ‘early 1990s’).
- The form of *non-consuming reference* represents temporal information that is not observable in the text, but it is assumed to be provided in other ways. For example, when a document is written on October 12 but this is not explicitly written in the document, then the Document Creation Time (e.g., ‘October 12’) can be given as meta-data. There are several kinds of such meta-data including Document Creation Time, Document Modification Time, Document Access Time, and others.

A temporal expression takes one of the above five forms, so it is necessary to convert it into a more structured template in order to use it for further applications. Given the sentence “Hoyeon and Younseo had breakfast at 9 o'clock”, there is the temporal expression ‘9 o'clock’ which should be converted into a structured form comprehensible by the computers. The structured form must represent the *extent*, *value*, and any additional information in the temporal expression. The *extent* is used to describe the position of the temporal expression in the raw text. For example, the position of the temporal expression ‘9시 [9 si]’ can be represented by indicating offset boundaries, where the offset boundary can be represented using a

token index or a character index. The *value* of the temporal expression is used to represent the temporal points (e.g., ‘2010-03-08’) or periods (e.g., ‘3 months’, ‘2 days’). It is worth noting that the same *value* can be represented by various expressions in text. For example, the temporal expressions ‘9 o'clock’ and ‘9:00AM’ denote the same value. This is the reason that temporal expressions must be converted into structured forms. The structured form should have a way of representing other information, such as temporal patterns (e.g., ‘two times a week’), indication of the temporal information type (e.g., DATE, TIME, DURATION), and indication of whether the temporal information is vague or not.

It is also necessary to convert the event expression into a structured template. For the sentence “Younseo eats a cookie.”, there is the event expression ‘eats’, where its structured form must represent the *extent*, the *class*, and some additional information of the event expression. The *extent* is used to describe the position of the event expression in the raw text. The *class* is used to represent the type of event. For example, the event expression ‘eats’ is the behavioral event experienced by ‘Younseo’, so the type of the event expression can be denoted as OCCURRENCE. The structured form of event expression should have a way to represent other information, such as polarity of the event, tense of the event expression, and so on.

Based on structured forms of temporal expression and event expression, a structured temporal relation between them can be generated. The relation must have two corresponding arguments and a relation type. For the sentence “Younseo eats a cookie at 9 o'clock.”, the two arguments of the relation are ‘9 o'clock’ and ‘eats’, and its relation type can be denoted as INCLUDES, so it means that the event ‘eats’ occurs at ‘9 o'clock’. The structured forms of temporal information must convey the core information of the temporal expression, the event expression, and the relation. Because the structured forms will be used in further applications, it is important to design forms that are effective and efficient. A package of structured forms is called *annotation language*, because it is used to annotate the raw text.

2.3 Temporal Information Extraction

As described earlier, for further application, temporal information must be converted into a structured form comprehensible by computers. The conversion process is a task of *temporal information extraction*. Because temporal information is useful for many applications, it is important to develop effective methods for extraction of temporal information.

The task of temporal information extraction strongly depends on the annotation language, because it will not be possible to extract temporal information that is not defined by the annotation language. In other words, the task can be defined as extraction of all the temporal information defined by the annotation language, but the task cannot be extraction of temporal information not defined by the annotation language. Different applications may adopt different parts of the temporal information, and they may introduce additions to the annotation language in order to achieve their final goals. As the annotation languages may not consider some language-specific characteristics, it will be necessary to revise the annotation languages to apply them to a target language.

The task of temporal information extraction is part of a larger application (e.g., QA systems, IR systems), so it is important to clarify the boundary of the task of temporal information extraction. Recall that the definition of temporal information is “information that can be used to order events or to measure the durations or intervals of events”. The events can be related to other tasks, such as spatial information extraction, subject-predicate-object (SPO) extraction, or sentiment prediction. That implies that the same

information might be extracted in more than one task, which may harm the overall efficiency of the application. Thus, it is important to set an appropriate boundary of the task of temporal information extraction.

There are three main approaches to the task of temporal information extraction: **rule-based**, **data-driven**, and **hybrid**. The rule-based approach is to define a set of rules, while the data-driven approach is to design an algorithm and define a set of features. The hybrid approach combines the rule-based approach and the data-driven approach. It is important to determine which approach will be used for extracting which part of the temporal information.

3. Task Boundary

Although there have been many studies related to the task of temporal information extraction, most of them did not clearly define their task boundaries. In this section, a structured way of describing the task boundary for temporal information extraction is proposed.

The task boundary of temporal information extraction can be determined using three sub-boundaries: a *boundary of temporal expressions*, a *boundary of event expressions*, and a *boundary of temporal relations*. As defined earlier, there are five forms of temporal expressions: *explicit* reference, *implicit* reference, *relative* reference, *vague* reference, and *non-consuming* reference. The boundary of temporal expressions is used to indicate the forms of the temporal expressions that are supposed to be extracted. If a desired system of temporal information extraction has the boundary of explicit reference, then the desired system will give only the temporal information derived from the temporal expressions taking the form of explicit reference. The event expressions are typically verbs or nouns. The boundary of event expressions indicates which one (e.g., verbs, nouns, or both) to extract. There are three kinds of boundary of temporal relations: a *kind boundary*, a *text boundary*, and a *transitivity boundary*. The kind boundary can contain at least one of three kinds: temporal links, subordinated links, and aspectual links. The temporal links are usually annotated by a *tlink* tag. The *tlink* tag can be either of *timex3-timex3* link (TT *tlink*), *timex3-makeinstance* link (TM *tlink*), *makeinstance-makeinstance* link (MM *tlink*), or a link between Document Creation Time and *makeinstance* (DM *tlink*). The subordinated links are usually annotated by a *slink* tag, while the aspectual links are typically annotated by an *alink* tag.

The second kind of temporal relation, namely the text boundary, indicates how many sentences/paragraphs/documents to consider for temporal relation extraction. For example, the temporal relation can be extracted for each sentence independently, or it can be extracted by considering two or more adjacent sentences. The text boundary can be one among the following options: single sentence, multiple sentences, single paragraph, multiple paragraphs, single document, or multiple documents. The local implicit reference of temporal expressions requires consideration of the temporal information obtained from all sentences that appeared before the target sentence, so one may argue that the text boundary always must be the single document when the boundary of temporal expressions contains the local implicit reference. However, the text boundary indicates whether the inter-sentence temporal relations are allowed or not, while the local implicit reference is just about the normalized values of temporal expressions, not about the inter-sentence temporal relations. Thus, the text boundary is independent of the boundary of temporal expressions.

The third kind of temporal relation, namely the transitivity boundary, indicates whether the transitivity in Allen's interval algebra is adopted or not, and indicates how many sentences will be processed with the

transitivity. The transitivity boundary is determined to one of the following options: none, single sentence, multiple sentences, single paragraph, multiple paragraphs, single document, or multiple documents. If the transitivity boundary is 'none', then there will be no temporal relations inferred using the transitivity in Allen's interval algebra. If the transitivity boundary is 'single sentence', then the transitivity will be applied to each sentence independently. For example, when the event e_1 occurred when time t_1 , and the event e_2 occurred before t_1 , then it can be inferred that e_2 occurred before e_1 although there is no expression representing the relation between e_1 and e_2 . If the transitivity boundary is larger than 'single sentence' (e.g., multiple sentences, single document), then there will be more temporal relations inferred using transitivity. Thus, it is necessary to determine the transitivity boundary carefully. When the transitivity boundary is either 'none' or 'single sentence', and the text boundary is 'multiple sentences' or larger boundary, then only explicit inter-sentence temporal relations will be extracted. That is, the inter-sentence temporal relations will be extracted only when there is at least one explicit temporal expression (e.g., '그 후 [Geu hoo]' (thereafter)). For example, if there are two sentences "He opened the door and came in." and "Thereafter, he slept.", then the inter-sentence temporal relation must be the linkage between the event 'came in' and the event 'slept'. If the transitivity boundary is two sentences (multiple sentences), then there will be one more inter-sentence temporal relation between 'opened' and 'slept'. When the kind boundary contains temporal links (e.g., *tlink* tags), it is also necessary to consider the types of *tlink* tags (e.g., TT *tlink*, TM *tlink*, etc.). This is called a *transitivity boundary for types of temporal links*.

Table 1. Summary of the task boundary of temporal information

| Task boundary | Targets |
|--|--|
| Temporal expressions (multiple choice) | Explicit reference Implicit reference (i.e., global implicit reference, local implicit reference) Relative reference Vague reference Non-consuming reference |
| Event expressions | Verbs Nouns Both of verbs and nouns |
| Temporal relations | |
| Kind boundary (multiple choice) | Temporal links Subordinated links Aspectual links |
| Text boundary | Single sentence/multiple sentences Single paragraph/multiple paragraphs Single document/multiple documents |
| Transitivity boundary | None Single sentence/multiple sentences Single paragraph/multiple paragraphs Single document/multiple documents |
| Transitivity boundary for types of temporal links (only available when kind boundary contains temporal links) | None TT <i>tlink</i> TM <i>tlink</i> MM <i>tlink</i> DM <i>tlink</i> |

To summarize, the boundary of temporal information can be summarized in Table 1. It is important to determine these sub-boundaries of temporal information before starting development of a system for extraction of temporal information.

4. History of Annotation Languages and Shared Tasks

The history of annotation languages and shared tasks can be summarized as shown in Fig. 3, where the orange dots represent the annotation languages and the blue dots denote the shared tasks. One notable thing is the appearance of Time Mark-up Language (TimeML) in 2003, which became the basis for many studies on extraction of temporal information. From 2007 to 2013, TempEval, which is a series of shared tasks, triggered many studies because it provided a high-quality dataset constructed using TimeML. The standardized version of TimeML, namely ISO-TimeML, appeared in 2009, and was revised in 2012. Between 2009 and 2011, some variations of TimeML were proposed as adaptations to particular languages (e.g., Korean, Italian).

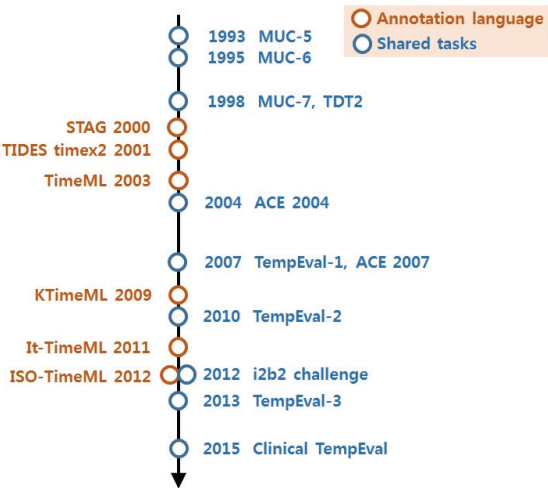


Fig. 3. History of annotation languages and shared tasks.

4.1 Shared Tasks

There were several shared tasks intended to develop systems for temporal information extraction from text. In the MUC-5 which was held in 1993, where there was a sub-task of assigning a calendrical time to a joint venture event [6]. At MUC-6 which was held in 1995, there was a sub-task of extraction of absolute temporal value as a part of the general task of Named Entity (NE) extraction [7]. The NE extraction task included the tag elements: *enamex* (for entity names, comprising organizations, persons, and locations), *timex* (for temporal expressions, namely direct mentions of dates and times), and *numex* (for number expressions, consisting only of direct mentions of currency values and percentages). As the proportion of *timex* tags in the test set was only 10%, the temporal information extraction was not the main part of the NE extraction task. The next relevant conference was held in 1998, namely MUC-7, extended the boundary of the sub-task to the extraction of relative temporal value [8].

In the field of Topic Detection and Tracking (TDT), the task of temporal information extraction became important because topic tracking is strongly related to the task of finding temporal relations between events. Since the shared task TDT-2 in 1998, there have been studies about extracting temporal information and applying it to final goals [9-11].

Based on TimeML, a series of shared tasks appeared, namely TempEval. The TempEval-1 was held as a task 15 of SemEval in 2007 [12], where it provided a dataset TimeBank constructed using the TimeML. There are three sub-tasks in TempEval-1: (1) the extraction of events and relations between them, (2) the extraction of events and relations with Document Creation Time, and (3) the extraction of temporal relations between major events in different sentences.

Also, in 2007, Automatic Content Extraction (ACE) opened recognition tasks related to the temporal information processing: the extraction of temporal expressions and events. These tasks have structures quite different from those of TempEval-1. For example, ACE 2007 has eight event types: life, movement, transaction, business, conflict, contact, personnel, and justice, which are completely different from those of TimeML.

The TempEval-2 was held as task 13 of SemEval in 2010 [13], where it provided datasets for six languages: Chinese, English, French, Italian, Spanish, and Korean. There are six sub-tasks in TempEval-2: (1) the extraction of *timex3* tags and their attributes, (2) the extraction of *event* tags and the attributes of *makeinstance* tags, (3) the extraction of temporal relations between *makeinstance* and *timex3* within the same sentence, (4) the extraction of temporal relations between *makeinstance* and Document Creation Time, (5) the extraction of temporal relations between major *makeinstance* tags of adjacent sentences, and (6) the extraction of temporal relations between two *makeinstance* tags.

TempEval-3 was held as a task 1 of SemEval in 2013 [14], where it provided datasets for English and Spanish. There are five sub-tasks in TempEval-3: (1) the extraction of *timex3* tags and their attributes, (2) the extraction of *event* tags and the attributes of *makeinstance* tags, (3) the extraction of all the tags from the texts, (4) the extraction of temporal relations given the correct *timex3*, *event*, and *makeinstance* tags, and (5) the extraction of temporal relation types given the correct argument pairs. In TempEval-3, for the task of extraction of *timex3*, the best performance was 77.61% (F1-measure) achieved by HeidelTime-t [15], which is a rule-based system. For the task of extraction of *event* and *makeinstance* tags, the best performance was 81.05% (F1-measure), and was achieved by ATT-1 [16] utilizing Maximum Entropy. For the task of extraction of *mlink* tags given correct other tags, the best performance was 36.26% (F1-measure) achieved by ClearTK-2 [17], using support vector machine (SVM) [18,19] and Logit. For the task of extraction of *mlink* tags without correct other tags, the best performance was 30.98% (F1-measure), achieved by ClearTK-2. As the state-of-the-art performance of temporal information extraction is not satisfactory, many researchers have kept trying to achieve better performance on this task.

There are shared tasks of temporal information extraction in the medical field. The Informatics for Integrating Biology and the Bedside (i2b2) offered a natural language processing (NLP) challenge in 2012 [20]. The goal of the i2b2 shared task was to develop a system for extracting temporal information from the discharge summaries of hospitals, where the temporal information is represented in a way similar to that for TimeML (e.g., *timex3*, *event*, *mlink*). The *mlink* tag of i2b2 has only three relation types: BEFORE, AFTER, and OVERLAP. Another shared task in the medical field is Clinical TempEval which was held as a task 6 of SemEval in 2015 [21], for which the goal was to develop a system for extracting temporal information from clinical texts. The temporal information was annotated with a new annotation language

modified from TimeML, because the temporal information in the medical field has some characteristics different from the general temporal information defined by the traditional TimeML.

4.2 Annotation Languages

As the task of temporal information extraction has become more important for many applications (e.g., QA systems, IR systems), it also has become important to design a language for annotating or representing temporal information. In TIDES (Translingual Information Detection, Extraction, and Summarization) supported by DARPA, introduced a *timex2* guideline [22] in 2000, in which the temporal values are represented in ISO-8601 [23]. Since then, *timex2* has evolved through several versions from 2001 to 2005. Similar to *timex*, the *timex2* is based on inline annotation. Based on the TIDES *timex2* guideline, there was a task of extraction of temporal information in ACE program in 2004, where the task includes the extraction of temporal expressions and prediction of temporal values.

TimeML was introduced as a new well-organized annotation language in 2003 [24]. It was mainly based on three previous works: TIDES *timex2* guideline, Sheffield Temporal Annotation Guidelines (STAG) [5], and another emerging work [25]. The TimeML was the first stable annotation language that incorporated temporal expressions, event expressions, and temporal relations.

As more studies appeared based on the TimeML, a standardized version of it, namely ISO-TimeML [26], was proposed in 2009, and revised in 2012. The ISO-TimeML has many parts in common with TimeML, but also has some additional tags and attributes. Many studies were based on the traditional TimeML adopted by the TempEval series, so the TimeML is the de facto standard while the ISO-TimeML is the de jure standard. To achieve generalization, the ISO-TimeML allows for modification of some parts of it, based on some language-specific characteristics. In [27], the Italian TimeML (It-TimeML), which is based on the ISO-TimeML was proposed, and it demonstrated the reliability of the It-TimeML guidelines and specifications based on the inter-coder agreement. The TimeML and ISO-TimeML might be stable and well organized, but language diversity was not well considered. For example, it was assumed that annotation is performed at a token level, which is not acceptable for some languages (e.g., Korean, Chinese). To overcome this limitation, Korean TimeML (KTimeML) was proposed as a new annotation language for Korean in 2009 [28]. It might be a solution to the limitation, but it has its own limitations. In [29], the limitations of the KTimeML are described, and a new revised version of KTimeML is proposed to address the limitations.

5. Temporal Information Extraction Methods

Because documents typically contain temporal information, many researchers have been attracted to developing systems for extracting such information from text. In 1972, a formal model for temporal references was presented [30]. In this study, a specific time was represented as an ordered pair whose elements are time points, so the temporal reference could be seen as a temporal relation between two time points. A better-structured definition of temporal relation was proposed in 1983 [31], of which the proposed 13 relation types are summarized in Table 2. Here, 13 relation types could represent every temporal relation between events, and it provided motivation for many additional studies related to temporal relation extraction.

Table 2. The Allen's 13 base temporal relations [31]

| Relation | Interpretation |
|-------------------|--|
| $X < Y$ | X takes place before Y |
| $Y > X$ | Y takes place after X |
| $X \text{ m } Y$ | X meets Y (X ends with beginning of Y) |
| $Y \text{ mi } X$ | X meets Y (<i>inversed notation</i>) |
| $X \text{ o } Y$ | X overlaps with Y |
| $Y \text{ oi } X$ | X overlaps with Y (<i>inversed notation</i>) |
| $X \text{ s } Y$ | X starts Y |
| $Y \text{ si } X$ | X starts Y (<i>inversed notation</i>) |
| $X \text{ d } Y$ | X during Y |
| $Y \text{ di } X$ | X during Y (<i>inversed notation</i>) |
| $X \text{ f } Y$ | X finishes Y |
| $Y \text{ fi } X$ | X finishes Y (<i>inversed notation</i>) |
| $X = Y$ | X is equal to Y |

Since the appearance of Allen's 13 temporal relations, there have been studies that were mainly based on linguistic assumptions or manually defined constraints. [32] utilized *Narrative convention* which is an assumption that the events of the current sentence must have occurred after the events of previous sentences. This assumption is simple, but can be effective for particular domains (e.g., stories). [33] proposed a manually defined set of rules to order the sequence of clause pairs. In [34] a new system for ordering events was proposed by analyzing the tense of the events. [35] used a linguistic model to incorporate temporal information for representing events. [36] extended the work of [33] by adding additional rules for ordering events in ambiguous cases. [37] analyzed how the compositionality affects the interpretation of temporal information, especially in the case of subordinated events. [38] proposed a method to label events with time periods based on TOODOR [39]. This method represented time values using eight units (e.g., day, century), while the time periods were represented by three types (e.g., time point, time interval, and span of time). The proposed method employed syntactic/semantic parsers, and used a set of rules for labeling the events. All of these studies commonly aimed at developing systems for ordering events by extracting the temporal information, where the proposed methods were mainly assumptions or constraints that were manually defined, based on linguistic knowledge or observations from the texts.

5.1 From TIMEX2 Scheme Looks old. Try to explore the recent developments, track the systems/research that are using timex2 and timex3. This is a part of NER at times.

Since the appearance of the TIDES *timex2* guideline in 2001, it became easier to generate/share a dataset because the guideline helped the datasets be consistent. This eventually led researchers to attempt to apply not only rules or linguistic constraints, but also machine-learning methods and mathematical models. In [40,41], systems for extracting relative temporal expressions and temporal relations between the events were proposed. They defined lexical rules by hand, and extended the rules automatically by a machine-learning method. In [3], it was assumed that there are three types of temporal expressions: explicit reference, indexical reference, and vague reference. The temporal expressions were extracted using finite state transducer (FST), and the event expressions were extracted using rules. The temporal relations were recognized as one of seven relation types (e.g., BEFORE, AFTER, INCLUDE, AT). [9] proposed a method

for extracting temporal information for TDT. The temporal expression candidates were extracted using finite state automata (FSA), and some of them were filtered out using a predefined dictionary. The absolute values of the recognized temporal expressions were extracted using a lexicon set and a set of rules.

5.2 From TimeML Scheme

Since TimeML appeared in 2003, some studies proposed annotation tools based on it, and some studies reported several limitations of TimeML and insisted that the TimeML should be changed. [42] explained how the TimeML was designed, and described some challenging issues and directions for future study. Two annotation tools, TANGO [43] and Callisto [44], were proposed that followed TimeML. [45] suggested adding a new tag CLINK to the TimeML. This study also insisted that there must be a function for denoting arguments in *event* tag. In [46], a tool for annotating temporal information was proposed, namely T-BOX, where the annotation was based on the TimeML. This presents events in temporal order. For example, when the event e_1 occurs before event e_2 , then e_1 is shown to the left of e_2 . Meanwhile, as the size of the cumulated dataset got larger, more studies attempted to use various machine-learning methods. Evita (Events In Text Analyzer) was proposed in [47], and was developed using the TARSQI framework [48]. Evita combines a statistical method and a set of rules to extract events and attributes (e.g., tense, aspect, modality, polarity, event class). In [49], a method for extracting temporal information from Chinese text was proposed. It defined a set of rules and utilized a chart-parsing based on constraints. [50] proposed a method for extracting temporal information from Swedish texts, and used the extracted temporal information to generate animated 3D scenes. It utilized finite state machine (FSM) and rules for extracting temporal expressions and event expressions. The temporal relations between events were extracted using decision trees (DT).

5.3 From TempEval and TempEval-2 Shared Tasks

Since TempEval, which is the well-known series of shared tasks, emerged in 2007, many studies mainly aimed at one or more sub-tasks defined by the TempEval. The publicly available dataset, namely TimeBank, was provided by TempEval. In [51], a method for extracting temporal relations between two events was proposed. It had two stages: (1) a machine-learning model for classifying event attributes (i.e., tense, aspect, modality, polarity, and event class), and (2) a machine-learning model for classifying the relation types between two events. It used TimeBank for experiments, and reported that Naive Bayes (NB) generally gives better performance than maximum entropy (ME). [52] proposed a method focused on the extraction of temporal expressions. It adopted the method of begin-inside-outside (BIO) tags, which are independent of the lengths of text segments. Poveda's method utilized TnT tagger [53] and YamCha toolkit [54], and compared the performance of SVM and FOIL (first-order inductive learner). It was reported that SVM generally gave better performance. In [55], a method for resolving conflicts between temporal relations was proposed. It used integer linear programming (ILP), and applied the transitivity assumption to generate additional relations or to remove inconsistent relations. After experiments with TimeBank, it was reported that the proposed method increased the performance accuracy by 3.6%. [56] proposed a method for extracting temporal relations between events and/or Document Creation Time. This study utilized Markov logic and defined a set of rules for Markov logic network (MLN).

The TempEval has been held triennially, and the TempEval-2 was held in 2010. During the three years from 2010 to 2012, a huge number of brilliant studies appeared. Many studies attempted to use various machine-learning methods, and some studies were about visualization of temporal information. Few studies provided reviews or surveys about temporal information extraction, and several studies tried to utilize patterns between temporal information and other information (e.g., spatial information). In [57], a new method for extracting events and temporal expressions was proposed. To extract events, it converted the results of TRIPS parser into a logical form, and used a set of rules defined using the logical forms. It also employed MLN to extract major events, and conditional random fields (CRF) for extracting temporal expressions. To predict the absolute temporal values, a set of manually defined rules was used. [58] proposed a system for ordering events and spatial information. It was based on an assumption that temporal information and spatial information appear within a particular distance (e.g., a sentence or paragraph). It first collected Wikipedia featured articles using unstructured information management applications (UIMA) [59], and attempted to extract spatial information using Meta-Carda GeoTagger [60], while the temporal information was extracted using a set of manually defined rules. In [61], a new corpus for the task of extraction of temporal expressions, namely WikiWars, was introduced. The source documents are collected from Wikipedia, and the annotation is performed using *timex2*. HeidelbergTime was proposed in [15], where it was found to be the best method for extracting temporal expressions in TempEval-2. It is a rule-based system that is portable, because it is based on UIMA. TimeTrails was introduced in [62], where its purpose was to help document analysis by visualizing the extracted temporal/spatial information. For this purpose, HeidelbergTime was employed to extract temporal information, and MetaCarda GeoTagger was used to extract spatial information. [63] employed Evita [47] and GUTime to extract temporal expressions and event expressions, and then used MLN to extract temporal relations. It also defined temporal entropy (TE) for evaluating the tightness of the extracted information within each document. TIPSem (Temporal Information Processing based on Semantic information) was proposed in [64], where it was one of the best methods in TempEval-2. CRF was used to extract temporal expressions and event expressions. This showed that using the semantic information conveying relations between elements could help with extraction of temporal information. Timely YAGO (T-YAGO) was proposed in [65], and is an extension of YAGO achieved by incorporating temporal aspects. Using this approach, temporal facts were extracted from Wikipedia infoboxes, categories, and lists. The extracted facts were integrated into the KB of T-YAGO.

In [66], a review of the current research trends was provided. This review included a number of applications that could benefit from temporal information, and discussed challenging issues. [67] used the expectation maximization (EM) algorithm to extract three types of temporal relations (BEFORE, AFTER, and OVERLAP). This was the first study that employed the EM algorithm for this task. In the E-step, the algorithm finds conflicts between the relations using a set of rules, and it replaces inconsistent relations using the probability values of the clusters, where each cluster is regarded as a relation type. In the M-step, the algorithm applies a smoothed relative-frequency estimate. In [68], PRAVDA was proposed, by which temporal facts could be automatically harvested from web text. For this, a pattern-based approach was used to extract candidate temporal expressions, and a label-propagation approach was employed to compute confidence scores of the candidates. In [69], YAGO2 was proposed. The purpose of this study was to extend the previous YAGO system [70] by incorporating temporal/spatial information. For this purpose, 5-tuple SPOTL (subject, predicate, object, time, and space), an extension from the 3-tuple SPO of YAGO, was used. This method extended KB by extracting the temporal/spatial

information from Wikipedia documents and WordNet [71]. For representation of the temporal information, it followed ISO-8601, while it used GeoNames to represent the spatial information. This showed that temporal/spatial information could be used to help extract facts from text more accurately.

In [72], an extended version of PRAVDA [68] was proposed. This combined the label propagation approach and an integer linear program, which eventually detects noisy events by incorporating temporal constraints among the events. SUTime, proposed in [73], was used for extracting temporal expressions and predicting temporal values. It is a part of the Stanford CoreNLP pipeline. In [74], a system for extracting temporal relations was proposed that took only six types of TimeML: SIMULTANEOUS, BEGINS, ENDS, BEFORE, IBEFORE, and INCLUDES. Herein, it used bootstrapped cross-document classification (BCDC) which takes additional relevant documents selected by the INDRI system [75] to re-train SVM models already trained using other training documents. The EM algorithm of [67] was adopted for extracting temporal relations. It was reported that the BCDC method worked well when the size of the dataset was small, and that the EM algorithm worked well when it was properly initialized. This implies that the proposed system works poorly with a biased dataset. [76] proposed a system for extracting temporal information from Wikipedia documents. It extended [65] by adding some named events to higher-order and first-order facts of T-YAGO. For this, it utilized a set of rules to extract temporal information from infobox, categories, titles, and lists of Wikipedia documents. Its usefulness was demonstrated by experimental results that it extracted 2-3-times more temporal facts and 50-times more events than T-YAGO and YAGO2 [69]. In [77], a survey about temporal information processing was provided. The report first introduced previous studies on information extraction, and described classical work in temporal information extraction and temporal reasoning. This work also provided research issues concerning the task of temporal information extraction, and listed some real-world applications.

5.4 From TempEval-3 Shared Task

As for TempEval-2, there have been many studies since TempEval-3 was held in 2013. Some studies attempted to find a way to effectively apply the temporal information to further applications (e.g., QA systems, KB systems), and few studies tried various machine-learning models as feature-generation models or classifiers. In several studies motivated from i2b2 challenges, systems of temporal information extraction were developed in another domain (e.g., clinical domain).

In [78], a system of temporal information extraction in the clinical domain was proposed. The goal of this study was to extract *timex3*, *event*, and *mlink* tags, from clinical texts. CRF was used to extract *event* tags, while *timex3* tags were extracted using a set of rules. Based on the extracted *timex3* and *event* tags, it extracted some *mlink* candidates using several rules. The candidates were filtered out using machine-learning methods (e.g., CRF, SVM). Another system in the clinical domain was proposed in [79], where its goal was to extract *timex3* tags and *event* tags from clinical texts. It made use of a set of hand-crafted rules for *timex3* extraction, and used the integer quadratic program (IQP) to infer attributes of *event* tags based on the assumption that the relations between two events might guide the inference procedure to determine the attributes of the other events. [80] proposed a method to predict temporal values from texts, for which it utilized context-free grammar (CFG) and rules. In [81] a system for temporal information extraction from clinical narratives was proposed. Its purpose was to extract *timex3* and *event* tags, which was basically a part of the i2b2 challenges. For this purpose, it employed HeidelTime [15] to

extract general *timex3* tags, and used a CRF-based sequence labeling method to extract domain-specific *timex3* tags and *event* tags.

A survey of temporal IR and related applications was provided in [82]. Although it focused on studies of temporal retrieval, it also discussed the task of temporal information extraction from Web documents. [83] attempted to extract temporal expressions and to find temporal values employing hand-engineered Combinatory Categorical Grammar (CCG). For this, context information (e.g., Document Creation Time, verb tense) was utilized to find the absolute values of temporal expressions. [84] proposed a system for populating KB by incorporating newly extracted temporal information. [85] proposed a sieve-based temporal ordering method, where the sieve represents a classifier. The sieve-based method is cascade architecture, such that each sieve passes its temporal relation decisions on to the next sieve. It was reported that the most precise sieves were collections of handcrafted rules, and insisted that the reason for this is that the intuition behind the rules is not easily captured by machine-learning models. [86] proposed a system for Korean language, which included a combination of machine-learning models and handcrafted rules. It incorporated a feature-generation model, namely the Language Independent Feature Extractor (LIFE) [87], to generate complementary features to improve the performance of the system.

In SemEval-2017, ‘Task 12: Clinical TemEval’ was held as a shared task for capturing temporal information on the clinical domain [88]. While notes from colon cancer patients had been used for both training and testing in Clinical TempEval 2015 and 2016, the training process used colon cancer patients data and the testing process used brain cancer patients data in Clinical TempEval 2017. [89] proposed a GUIR model that combines CRFs and decision tree ensembles constructed on lexical features (e.g., uppercases, lowercases, prefixes, suffixes, punctuations, stop words, etc.), rule-based features for complex patterns or specific words, and distributional features (e.g., word clusters and word embeddings). [90] proposed Hitachi model that combines CRFs, neural networks and decision tree ensembles trained by lexical features (e.g., n-grams of nearby words, character n-grams, prefixes, suffixes, etc.) and common features (e.g., POS tags, verb tenses, sentence lengths, event/time tokens, number of other event/time mentioned, etc.). [91] proposed KULeuven-LIIR model that combines SVMs for detecting event/time expressions and a structured perceptron for temporal relations. [92] proposed LIMSI-COT model that uses neural network-based methods to detect both intra and inter-sentence relations. As a result of the shared task, in the case of time span extraction, GUIR model showed the best performance with 0.57 F_1 score, and in the case of extracting the time span and class together, KULeuven-LIIR model showed the best result with 0.53 F_1 score. Although LIMSI-COT model showed relatively low results with the F_1 score than others, it had the highest result of recall with 0.66 and 0.63 for each case. These results demonstrate that models considering various rules and grammatical elements are more suitable for finding time spans than others such as neural network-based models.

In SemEval-2018, ‘Task 6: Parsing Time Normalizations’ was held as a shared task related to time information extraction [93]. The purpose of this task is to develop new techniques that allow time normalization based on recognizing semantically compositional time operators. For this task, they presented two tracks—identifying the time operators, and providing time intervals on the timeline. Here, the compositional time operators are underlying the proposed method in previous work [94]. Olex et al. [95] submitted a Chrono model that applies the rule-based approach as primary and a Chrono* model that improves some bugs on their previous model. Their models have captured temporal tokens with temporal expressions or regular expressions for specific words, and connected some consecutive tokens to find temporal phrases.

5.5 Summary

To summarize, many studies focused on either a rule-based approach or a data-driven approach, or both of them. It seems that the most powerful approach for the task of *timex3* tag extraction is the rule-based approach, while the most powerful approach for the task of *event* tag extraction is the data-driven approach. In terms of the task of *mlink* tag extraction, the data-driven approach seems the best.

Many recent studies attempted to make use of machine-learning methods for the task of temporal information extraction. The features adopted from machine-learning methods can be summarized as follows. To extract *timex3* tags, given a particular window size W surrounding the target token, the features include n-grams of tokens, n-grams of POS tags, the top ontology class of WordNet, the frequencies of the target token, the suffix and prefix, whether n-grams are upper-case or not, whether n-grams are digits or not, whether the first character is upper-case or not, whether the previous token is a temporal expression or not, the head token of the target token, the semantic role label of the target token, and the semantic role labels of subordinated tokens. In particular, the attribute *value* of *timex3* tag was predicted primarily by a set of rules rather than by machine-learning methods. To extract *event* tags, their features were defined very similarly to *timex3* tags. The attributes of *makeinstance* tags (e.g., *polarity*, *modality*, *tense*) are predicted mainly by a set of rules. To extract *mlink* tags between *timex3* and *makeinstance*, the features include such items as n-grams of tokens of the argument tags, n-grams of POS tags of the argument tags, whether the two arguments are in the same sentence, the head preposition, and whether there is a temporal expression of interval nearby the *timex3* tag. To extract *mlink* tags between two *makeinstance* tags, the features include such as n-grams of tokens of the argument tags, n-grams of POS tags of the argument tags, the WordNet synset of the token of each argument, the verbs subordinated by the arguments, the adverbs attached to the verbs if the arguments are verbs, whether the arguments have the same *tense*, whether the arguments have the same *aspect*, a pair of *tense* of the two arguments, a pair of *aspect* of the two arguments, and a pair of *class* of the *event* tags related to the arguments.

As shown above, the defined features are heavily related to linguistic observations, so major effort is required to put a heavy effort to feature engineering process with consideration of language-specific characteristics. Most of the previous studies were focused on the use of English, so it is necessary to investigate the best way to extract temporal information using the Korean language.

6. Research Issues of Temporal Information Extraction

6.1 Perspective on Knowledge

The task of temporal information extraction has several research issues still unresolved. The first issue is the *design of annotation language* for specific purposes. The purpose might be a particular application (e.g., QA system) or a particular language having distinct characteristics that cannot be annotated with the existing annotation languages. It would be better to make the annotation language more general, so that it can be used to annotate any expression conveying temporal information. It should also incorporate language-specific characteristics of the target language. If an annotation language misses some language-specific characteristics, it might harm the performance of applications developed using the poor annotation language. Moreover, if there are some expressions that cannot be annotated with the annotation language,

then such temporal expressions will never be extracted by the system developed using the annotation language. This will eventually cause further applications to be deficient from the missing temporal information. Thus, the annotation language should be carefully designed.

The second issue is the *construction of dataset for each language/purpose*. Since the TimeML has appeared, there have been several other datasets that could be used for studies of temporal information processing. However, these are mostly English datasets. The datasets of other languages are relatively small and typically have many annotation errors because not enough time was taken by the founders of such datasets to consider completely the characteristics of their target languages. This issue is strongly related to the first issue, because the dataset will be poor without carefully designed annotation language for each target language. If the annotation language incorporates language-specific characteristics and is sufficiently expressive, then a high-quality dataset can be constructed using a part of the annotation language for its own purpose. For different purposes, different parts of the annotation language could be adopted to construct the dataset. For example, if the purpose is to develop an application that simply recognizes temporal expressions, then it will be sufficient that the dataset contains only *timex3* tags, without any other tags or attributes. It would be better, of course, if the dataset has all the tags and attributes defined by the annotation language. However, because manual annotation takes a great deal of time, it is necessary to determine which part of the annotation language to use for constructing the dataset, given consideration of the purpose.

The third issue is the *temporal context*. If the boundary of temporal expressions contains relative reference, then it is necessary to design an algorithm for maintaining the temporal context. Given the two sentences “Tommy was born in 1990.” and “After 10 years, he went to jail.”, it will be difficult to get the value of ‘After 10 years’ without considering the current time in the previous sentence. The current time for each sentence is called its temporal context, and it is not trivial to design an algorithm to maintain the temporal context. The simplest algorithm is just to update the temporal context when there is a temporal expression of explicit reference within the corresponding sentence. The first sentence of the example above has the explicit reference ‘1990’, so the temporal context can be updated to 1990. This algorithm may fail to track the temporal context in some cases. For example, if the two sentences above were followed by the sentence “After 2 years, he was released from the prison.”, then the correct value of ‘After 2 years’ must be 2002. However, the simplest algorithm will give 1992, because there is no explicit reference in the previous sentence, and the value 1990 obtained from the first sentence is the latest temporal context. Although the algorithm has such problems, it works well in most cases because this problem does not happen very often.

The fourth issue is the *temporary knowledge-base (TKB)*. If the boundary of temporal expressions contains local implicit reference, then it is necessary to design a method for maintaining the TKB. For the two sentences “Tommy was born 1990.” and “He went to jail when he was 10 years old.”, it is impossible to infer the value of ‘when he was 10 years old’ without using the temporal information extracted from the first sentence. This is different from the temporal context, because this case requires a kind of semantic reasoning. For example, it is required to know that the value of ‘when he was 10 years old’ can be computed based on the knowledge of when the event ‘born’ happened in the first sentence. Thus, such knowledge extracted from the local implicit reference must be maintained. The collection of the knowledge is defined as the TKB, where TKB can be maintained per paragraph, document, or even corpus. In most case, TKB will be maintained per document. The cost of maintenance of TKB per document will be expensive, so it will be beneficial to develop an efficient TKB.

The tenth issue is the external knowledge-base (EKB). If the boundary of temporal expressions contains global implicit reference, then it is necessary to design a method for communicating with an EKB. For the sentence “During the Koryo Dynasty, the ancestors developed it.”, it is impossible to obtain the value of ‘the Koryo Dynasty’ without using some external resources (e.g., KB). Such a KB is defined as an EKB, as it essentially does not belong within the boundary of temporal information extraction.

6.2 Perspective on Development

The first issue is the *development of annotation tools*. This issue is related to the second issue, because the annotation tools are supposed to be used to construct the datasets. This is also related to the first issue, because it must be determined which annotation language to use before development of the annotation tools is started. Given a particular annotation language, the annotation tools must satisfy three requirements. First, it should provide ways to annotate all the tags and attributes defined by the annotation language. Second, it should be easy for the annotators to use. This is about the interactions between humans and the tools. Third, it should be able to generate annotated files with at least one well-known format (e.g., XML format, JSON format). Well-developed annotation tools satisfying these requirements will make construction of datasets easier and faster.

The second issue is the *system structure*. As the task of temporal information extraction can be divided into several sub-tasks, it is necessary to determine how to design the structure of the system. For example, if the system has three sub-tasks: extraction of temporal expressions, extraction of event expressions, and extraction of temporal relations, then the system might have a cascade structure that conducts the three sub-tasks in order. Several factors must be considered in the design of the system structure. Some sub-tasks may be performed concurrently, and some sub-tasks may not be performed without the results from other particular sub-tasks. Some sub-tasks may benefit from the results of other sub-tasks. Furthermore, the system may require preprocessing (e.g., language analytic tools) or post-processing (e.g., result formatting). Thus, it is necessary to design the system structure with consideration of such factors.

The third issue is the *investigation of usefulness of various feature generators*. Given the text, raw features—e.g., part-of-speech (POS) tags, Named Entity (NE) tags, dependency structures—can be generated. Based on the raw features, higher-level features can be derived. For example, given a pair of two morphemes with their POS tags, a high-level feature could be an indication of whether their POS tags are the same or not. Because manual feature-engineering requires a great deal of time, several automatic feature generators were proposed, such as tree-kernel functions, deep neural networks, and probabilistic topic models. The tree-kernel functions require dependency parsing as preprocessing and are known to convey syntactic patterns of the text, so it could be useful for relation extraction. The deep neural networks and topic models typically do not require linguistic knowledge, and they generate real-valued vectors or integer values as features. They are known to convey semantic features or semantic/syntactic features. Other feature generation methods could also be considered.

The fourth issue is the *inter-sentence temporal relation extraction*. Many studies of temporal information extraction were focused on extraction from each sentence independently. That is, the text boundary of these studies is a ‘single sentence’. If the text boundary is larger than ‘single sentence’ (e.g., multiple sentences), then it is necessary to find a way to extract inter-sentence temporal relations. In such cases, it should be determined whether implicit inter-sentence relations (e.g., the relations inferred by transitivity) are extracted or not.

6.3 Other Perspectives

The first issue is the *investigation of ways for achieving high performance for each sub-task*. The task of temporal information extraction can be divided into several sub-tasks, and the best methods for the different sub-tasks will be different. Thus, it is necessary to find the best method for each sub-task. According to state-of-the-art research trends, a rule-based approach is best for extraction of temporal expressions, while a machine-learning approach is best for extraction of event expressions. For each approach, it is also necessary to find the most effective algorithm or model. For example, even if the rule-based approach turns out to be the best for a particular sub-task, it is still required to find the best set of rules. Similarly, in terms of the machine-learning approach, it is still necessary to find the best specific machine-learning model. This issue also includes the parameter settings of the models.

The second issue is the *harmony with other tasks*. Temporal information is probably combined with the results of some other tasks, such as spatial information extraction, co-reference resolution, or semantic role labeling. This issue might seem that it is not about the task of temporal information extraction, but there might be redundancy among the tasks unless this issue is considered. For example, if one uses the results of the task of semantic role labeling to help the task of temporal information extraction, then some of the predicted semantic roles might be the same as some of the extracted event expressions. Thus, it is necessary to find a way to avoid such redundancy, and to apply the semantic role labels effectively to the system of temporal information extraction.

The third issue is the *contradiction resolution of temporal relations*. There could be contradiction among the extracted temporal relations. For example, if event e_1 occurred before event e_2 , and e_2 occurred before event e_3 , then it is contradiction when there is a temporal relation in which event e_3 occurred before e_1 . This may happen often when the text boundary is greater than or equal to multiple sentences.

The fourth issue is the *definition of task boundary structure*. Although there were many studies about the task of temporal information extraction, there was no clear definition about the structure of the task boundary. Most of the existing studies relied heavily on the task definition provided by shared tasks (e.g., TempEval), but such task definition misses some aspects of the temporal information to be extracted. For example, the TempEval does not take transitivity into account as a task boundary, and the transitivity can be a serious factor for the temporal relation extraction.

The fifth issue is the *time zone*. Given a question “When a plane took off from the Incheon airport (South Korea) at 8:00AM and landed in Shanghai (China) at 9:00AM, what is the flight time?”, the QA system will give the answer ‘1 hour’ if it does not consider the time zone. However, the answer is wrong because there is a time lag of an hour between Incheon and Shanghai. To deal with this issue, it will be necessary to investigate a way to incorporate the time zone into the temporal information processing.

7. Evaluation Metrics

When a dataset is annotated, it is necessary to evaluate the quality of the dataset. This is usually performed using Cohen’s Kappa κ or Fleiss’s Kappa κ [96]. Cohen’s Kappa measures the agreement between two annotators, while Fleiss’s Kappa measures the agreement among three or more annotators. Greater Kappa values indicate that the corresponding dataset is annotated in a more consistent way, which in turn implies that the dataset is more reliable. More details of the Kappa values can be found in [96].

When a system for temporal information extraction is developed, it is necessary to evaluate the system. For the evaluation, the dataset is typically divided into a training set, validation set, and test set, where the validation set is used to find the best parameter setting and the test set is completely unseen until the system is tested. The evaluation could also be performed using other methods such as k-fold cross validation, hold-out cross validation, and leave-one-out cross validation.

When the dataset is prepared, it is necessary to determine which metric to use. There are a number of available metrics such as accuracy, general F_β score, ROC (receiver operating characteristic), and so on. Among the metrics, the most widely used ones are precision, recall, and F_1 score, which is computed by a combination of the precision and the recall. The formula of the F_1 score is as following Eq. (1).

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (1)$$

When the dataset is prepared and a particular metric is chosen, there are still several issues that must be considered during the evaluation. First, it must be determined how to evaluate the predicted extent of tags. The tag extent can be evaluated in a strict manner or a soft manner. For the strict manner, only perfectly predicted extents are regarded as correctly predicted tags, while for the soft manner, predicted extents with small errors are also regarded as correctly predicted tags. For the sentence “I will go there tomorrow morning.”, there is one *timex3* tag whose correct extent is ‘tomorrow morning’. If the system predicts that the extent of *timex3* tag is ‘tomorrow’, then it will be regarded as an incorrect prediction by the strict manner. On the other hand, when the soft manner with ‘1 token error’ is employed, then the tag extent ‘tomorrow’ is also regarded as a correct prediction. In most cases, the strict manner is used to measure the tag extents.

Second, it is necessary to determine whether the tag attributes will be evaluated independently or not. If the predicted *timex3* tag has two attributes *type* and *value*, then each attribute can be evaluated independently or evaluated in a sequential manner. The process for the sequential manner is that, given a particular order of attributes, the evaluation of each attribute is performed using only the tags with correctly predicted precedent attributes. For example, when the order of *timex3* attributes is the sequence of extent, *type*, and *value*, then the *type* prediction is evaluated using only the tags with correctly predicted extents. Thus, the performance of extent prediction will influence the performance of the following attributes (e.g., *type* and *value*).

Third, it must be determined whether the prediction of temporal relation tags (e.g., *tlink* tags) is performed using the other correct tags or not, where the other tags are *timex3*, *event*, and *makeinstance* tags. Because the temporal relation is a relation between two argument tags, there are two ways to evaluate the relation tags: (1) evaluation given the other correct tags and (2) evaluation given the other predicted tags. It would be best, of course, if both ways were performed.

8. Conclusion

Although the history of the field of temporal information extraction is short, there have been many studies related to this subject. In this paper, studies related to the temporal information extraction are discussed, and summarized in chronological order. The history of annotation languages and shared tasks is described, and some issues about the temporal information (e.g., task boundary, research issues,

evaluation metrics, and applications) are discussed. To summarize the trend of recent research about the temporal information extraction, many studies have been focused on applying various methods to the task of the temporal information extraction, because the size of the datasets to be handled, and machine-learning technologies, are developing rapidly. So far, the rule-based approach seems best for the task of *timex3* extraction, while the data-driven approach (e.g., CRF, SVM) seems best for the task of *event* and the task of *link* extraction.

In the future, it will be necessary to develop give an effort to rule/feature engineering in order to improve performance, and other machine-learning models should be designed or investigated. It is also necessary to find a way to combine wisely the rules and the machine-learning models, because the combination may result in synergetic effects.

Acknowledgement

This work was supported by Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2013-2-00131, Development of Knowledge Evolutionary WiseQA Platform Technology for Human Knowledge Augmented Services), and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2019021348).

References

- [1] M. Baldassarre, "Think big: learning contexts, algorithms and data science," *Research on Education and Media*, vol. 8, no. 2, pp. 69-83, 2016.
- [2] Wikipedia, "Time," [Online]. Available: <https://en.wikipedia.org/wiki/Time>.
- [3] F. Schilder and C. Habel, "Temporal information extraction for temporal question answering," in *New Directions in Question Answering: Papers from the 2003 AAAI Symposium*. Menlo Park, CA: AAAI Press, 2003, pp. 35-44.
- [4] O. Alonso, M. Gertz, and R. Baeza-Yates, "On the value of temporal information in information retrieval," *ACM SIGIR Forum*, vol. 41, no. 2, pp. 35-41, 2007.
- [5] A. Setzer and R. J. Gaizauskas, "Annotating events and temporal information in newswire texts," in *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC)*, Athens, Greece, 2000, pp. 1287-1294.
- [6] US Advanced Research Projects Agency, *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*. San Francisco, CA: Morgan Kaufmann, 1993.
- [7] "Named entity task definition (version 2.0)," 1995 [Online]. Available: https://cs.nyu.edu/cs/faculty/grishman/NEtask20.book_1.html.
- [8] N. Chinchor, "Appendix D: MUC-7 Information extraction task definition (version 5.1)," in *Proceedings of the 7th Message Understanding Conference (MUC-7)*, Fairfax, VA, 1998.
- [9] P. Kim and S. H. Myaeng, "Usefulness of temporal information automatically extracted from news articles for topic tracking," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 3, no. 4, pp. 227-242, 2004.
- [10] J. Allan, J. G. Carbonell, G. Doddington, J. Yamron, and Y. Yang, "Topic detection and tracking pilot study final report," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, 1998, pp. 194-218.

- [11] Y. Yang, J. G. Carbonell, R. D. Brown, T. Pierce, B. T. Archibald, and X. Liu, "Learning approaches for detecting and tracking news events," *IEEE Intelligent Systems and their Applications*, vol. 14, no. 4, pp. 32-43, 1999.
- [12] M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, J. Moszkowicz, and J. Pustejovsky, "The TempEval challenge: identifying temporal relations in text," *Language Resources and Evaluation*, vol. 43, no. 2, pp. 161-179, 2009.
- [13] M. Verhagen, R. Sauri, T. Caselli, and J. Pustejovsky, "SemEval-2010 Task 13: TempEval-2," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, Uppsala, Sweden, 2010, pp. 57-62.
- [14] N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen, and J. Pustejovsky, "Semeval-2013 task 1: TempEval-3: evaluating time expressions, events, and temporal relations," in *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*, Atlanta, GA, 2013, pp. 1-9.
- [15] J. Strötgen and M. Gertz, "HeidelTime: High quality rule-based extraction and normalization of temporal expressions," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, Uppsala, Sweden, 2010, pp. 321-324.
- [16] H. Jung and A. Stent, "ATT1: temporal annotation using big windows and rich syntactic and semantic features," in *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*, Atlanta, GA, 2013, pp. 20-24.
- [17] S. Bethard, "ClearTK-timeml: a minimalist approach to TempEval 2013," in *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*, Atlanta, GA, 2013, pp. 10-14.
- [18] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the 5th Annual Workshop on Computational Learning Theory*, Pittsburgh, PA, 1992, pp. 144-152.
- [19] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.
- [20] The Informatics for Integrating Biology and the Bedside (i2b2), "2012 NLP Shared Task," 2012 [Online]. Available: <https://www.i2b2.org/NLP/TemporalRelations/>.
- [21] S. Bethard, L. Derczynski, G. Savova, J. Pustejovsky, and M. Verhagen, "Semeval-2015 task 6: clinical TempEval," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval)*, Denver, CO, 2015, pp. 806-814.
- [22] L. Ferro, I. Mani, B. Sundheim, and G. Wilson, "TIDES Temporal Annotation Guidelines (version 1.0.2)," The MITRE Corporation, McLean, VA, 2001.
- [23] *Data elements and interchange formats – Information interchange –Representation of dates and times*, ISO 8601, 2004.
- [24] J. Pustejovsky, J. M. Castano, R. Ingria, R. Sauri, R. J. Gaizauskas, A. Setzer, G. Katz, and D. Radev, "TimeML: robust specification of event and temporal expressions in text," in *Proceedings of AAAI Spring Symposium on New Directions in Question Answering*, Stanford, CA, 2003, pp. 28-34.
- [25] G. Katz and F. Arosio, "The annotation of temporal information in natural language sentences," in *Proceedings of the Workshop on Temporal and Spatial Information Processing*, Stroudsburg, PA, 2001.
- [26] *Language resources management - Semantic annotation framework (SemAF) - Part1: Time and events*, ISO 24617-1:2012, 2012.
- [27] T. Caselli, V. B. Lenzi, R. Sprugnoli, E. Pianta, and I. Prodanof, "Annotating events, temporal expressions and relations in Italian: the It-TimeML experience for the Ita-TimeBank," in *Proceedings of the 5th Linguistic Annotation Workshop*, Portland, OR, 2011, pp. 143-151.
- [28] S. Im, H. You, H. Jang, S. Nam, and H. Shin, "KTimeML: specification of temporal and event expressions in Korean text," in *Proceedings of the 7th Workshop on Asian Language Resources*, Singapore, 2009, pp. 115-122.
- [29] Y. S. Jeong, W. T. Joo, H. W. Do, C. G. Lim, K. S. Choi, and H. J. Choi, "Korean TimeML and Korean TimeBank," in *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*, Portoroz, Slovenia, 2016, pp. 356-359.
- [30] B. C. Bruce, "A model for temporal references and its application in a question answering program," *Artificial Intelligence: An International Journal*, vol. 3, pp. 1-26, 1972.

- [31] J. F. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM*, vol. 26, no. 11, pp. 832-843, 1983.
- [32] D. R. Dowty, "The effects of aspectual class on the temporal structure of discourse: semantics or pragmatics?," *Linguistics and Philosophy*, vol. 9, no. 1, pp. 37-61, 1986.
- [33] B. L. Webber, "Tense as discourse anaphor," *Computational Linguistics*, vol. 14, no. 2, pp. 61-73, 1988.
- [34] R. J. Passonneau, "A computational model of the semantics of tense and aspect," *Computational Linguistics*, vol. 14, no. 2, pp. 44-60, 1988.
- [35] M. Moens and M. Steedman, "Temporal ontology and temporal reference," *Computational Linguistics*, vol. 14, no. 2, pp. 15-28, 1988.
- [36] F. Song and R. Cohen, "Tense interpretation in the context of narrative," in *Proceedings 9th National Conference on Artificial Intelligence*, Anaheim, CA, 1991, pp. 131-136.
- [37] C. H. Hwang and L. K. Schubert, "Tense trees as the "fine structure" of discourse," in *Proceedings of the 30th Annual Meeting on Association for Computational Linguistics*, Newark, DE, 1992, pp. 232-240.
- [38] D. Llido, R. Berlanga, and M. J. Aramburu, "Extracting temporal references to assign document event-time periods," in *Database and Expert Systems Applications*. Heidelberg: Springer, 2001, pp. 62-71.
- [39] M. J. Aramburu-Cabo and R. Berlanga-Llavori, "Retrieval of information from temporal document databases," in *Object-Oriented Technology: ECOOP 1999 Workshop Reader*. Heidelberg: Springer, 1999, p. 215.
- [40] I. Mani, B. Schiffman, and J. Zhang, "Inferring temporal ordering of events in news," in *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, Edmonton, Canada, 2003, pp. 55-57.
- [41] I. Mani and G. Wilson, "Robust temporal processing of news," In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, Hong Kong, China, 2000, pp. 69-76.
- [42] I. Mani, "Recent developments in temporal information extraction," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, Borovets, Bulgaria, 2003, pp. 45-60.
- [43] Tango - annotation tool [Online]. Available: <http://www.timeml.org/tango/tool.html>.
- [44] D. S. Day, C. McHenry, R. Kozierok, and L. D. Riek, "Callisto: a configurable annotation workbench," in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, Lisbon, Portugal, 2004.
- [45] J. Pustejovsky, J. Littman, and R. Sauri, "Argument structure in TimeML," in *Dagstuhl Seminar Proceedings*. Wadern, Germany: Schloss Dagstuhl, Leibniz-Zentrum für Informatik, 2006.
- [46] M. Verhagen, "Drawing TimeML Relations with TBox," in *Annotating, Extracting and Reasoning about Time and Events*. Heidelberg: Springer, 2007, pp. 7-28.
- [47] R. Sauri, R. Knippen, M. Verhagen, and J. Pustejovsky, "Evita: a robust event recognizer for QA systems," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Vancouver, Canada, 2005, pp. 700-707.
- [48] M. Verhagen, I. Mani, R. Sauri, R. Knippen, J. B. Jang, J. Littman, A. Rumshisky, J. Phillips, and J. Pustejovsky, "Automating temporal annotation with TARSQI," in *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, Ann Arbor, MI, 2005, pp. 81-84.
- [49] W. Mingli, L. Wenjie, L. Qin, and L. Baoli, "CTEMP: a Chinese temporal parser for extracting and normalizing temporal information," in *Natural Language Processing - IJCNLP 2005*. Heidelberg: Springer, 2005, pp. 694-706.
- [50] A. Berglund, R. Johansson, and P. Nugues, "A machine learning approach to extract temporal information from texts in Swedish and generate animated 3D scenes," in *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, Trento, Italy, 2006, pp. 385-392.
- [51] N. Chambers, S. Wang, and D. Jurafsky, "Classifying temporal relations between events," in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, Prague, Czech Republic, 2007, pp. 173-176.

- [52] J. Poveda, M. Surdeanu, and J. Turmo, "A comparison of statistical and rule-induction learners for automatic tagging of time expressions in English," in *Proceedings of the 14th International Symposium on Temporal Representation and Reasoning (TIME'07)*, Alicante, Spain, 2007, pp. 141-149.
- [53] T. Brants, "TnT: statistical part-of-speech tagging," 1998 [Online]. Available: <http://www.coli.uni-saarland.de/~thorsten/tnt/>.
- [54] T. Kudo, "YamCha: yet another multipurpose CHunk annotator," 2013 [Online]. Available: <http://chasen.org/~taku/software/yamcha/>.
- [55] N. Chambers and D. Jurafsky, "Jointly combining implicit constraints improves temporal ordering," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Honolulu, HI, 2008, pp. 698-706.
- [56] K. Yoshikawa, S. Riedel, M. Asahara, and Y. Matsumoto, "Jointly identifying temporal relations with Markov logic," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Singapore, 2009, pp. 405-413.
- [57] N. UzZaman and J. F. Allen, "Event and temporal expression extraction from raw text: first step towards a temporally aware system," *International Journal of Semantic Computing*, vol. 4, no. 4, pp. 487-508, 2010.
- [58] J. Strotgen, M. Gertz, and P. Popov, "Extraction and exploration of spatio-temporal information in documents," in *Proceedings of the 6th Workshop on Geographic Information Retrieval*, Zurich, Switzerland, 2010.
- [59] Apache Software Foundation, "Apache UIMA," 2013 [Online]. Available: <http://uima.apache.org/>.
- [60] Qbase, "MetaCarta," [Online]. Available: <http://qbase.com/products/metacarta/>.
- [61] P. Mazur and R. Dale, "WikiWars: a new corpus for research on temporal expressions," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, Cambridge, MA, 2010, pp. 913-922.
- [62] J. Strotgen and M. Gertz, "TimeTrails: a system for exploring spatio-temporal information in documents," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 1569-1572, 2010.
- [63] X. Ling and D. S. Weld, "Temporal information extraction," in *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, Atlanta, GA, 2010, pp. 1385-1390.
- [64] H. Llorens, E. Saquete, and B. Navarro, "TIPSem (English and Spanish): evaluating CRFs and semantic roles in tempeval-2," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, Los Angeles, CA, 2010, pp. 284-291.
- [65] Y. Wang, M. Zhu, L. Qu, M. Spaniol, and G. Weikum, "Timely YAGO: harvesting, querying, and visualizing temporal knowledge from Wikipedia," in *Proceedings of the 13th International Conference on Extending Database Technology*, 2010, pp. 697-700.
- [66] O. Alonso, J. Strotgen, R. A. Baeza-Yates, and M. Gertz, "Temporal information retrieval: challenges and opportunities," in *Proceedings of Workshop on Linked Data on the Web*, Hyderabad, India, 2011, pp. 1-8.
- [67] S. A. Mirroshandel and G. Ghassem-Sani, "Temporal relation extraction using expectation maximization," in *Proceedings of the International Conference Recent Advances in Natural Language Processing*, Hissar, Bulgaria, 2011, pp. 218-225.
- [68] Y. Wang, B. Yang, L. Qu, M. Spaniol, and G. Weikum, "Harvesting facts from textual web sources by constrained label propagation," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, Glasgow, Scotland, 2011, pp. 837-846.
- [69] J. Hoffart, F. M. Suchanek, K. Berberich, E. Lewis-Kelham, G. De Melo, and G. Weikum, "YAGO2: exploring and querying world knowledge in time, space, context, and many languages," in *Proceedings of the 20th International Conference Companion on World Wide Web*, Hyderabad, India, 2011, pp. 229-232.
- [70] M. S. Fabian, K. Gjergji, and W. Gerhard, "Yago: a core of semantic knowledge unifying WordNet and Wikipedia," in *Proceedings of the 16th International World Wide Web Conference*, Banff, Canada, 2007, pp. 697-706.
- [71] WordNet [Online]. Available: <https://wordnet.princeton.edu/>.

- [72] Y. Wang, M. Dylla, M. Spaniol, and G. Weikum, "Coupling label propagation and constraints for temporal fact extraction," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers*, Jeju, Korea, 2012, pp. 233-237.
- [73] A. X. Chang and C. D. Manning, "SUTime: a library for recognizing and normalizing time expressions," in *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey, 2012, pp. 3735-3740.
- [74] S. A. Mirroshandel and G. Ghassem-Sani, "Towards unsupervised learning of temporal relations between events," *Journal of Artificial Intelligence Research*, vol. 45, pp. 125-163, 2012.
- [75] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft, "Indri: a language model-based search engine for complex queries," in *Proceedings of the International Conference on Intelligent Analysis*, McLean, VA, 2005, pp. 2-6.
- [76] E. Kuzey and G. Weikum, "Extraction of temporal facts and events from Wikipedia," in *Proceedings of the 2nd Temporal Web Analytics Workshop*, Lyon, France, 2012, pp. 25-32.
- [77] I. Berrazega, "Temporal information processing: a survey," *International Journal on Natural Language Computing*, vol. 1, no. 2, pp. 1-14, 2012.
- [78] B. Tang, Y. Wu, M. Jiang, Y. Chen, J. C. Denny, and H. Xu, "A hybrid system for temporal information extraction from clinical text," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 828-835, 2013.
- [79] P. Jindal and D. Roth, "Extraction of events and temporal expressions from clinical narratives," *Journal of Biomedical Informatics*, vol. 46, pp. S13-S19, 2013.
- [80] S. Bethard, "A synchronous context free grammar for time normalization," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Seattle, WA, 2013, pp. 821-826.
- [81] Y. K. Lin, H. Chen, and R. A. Brown, "MedTime: a temporal information extraction system for clinical narratives," *Journal of Biomedical Informatics*, vol. 46, pp. S20-S28, 2013.
- [82] R. Campos, G. Dias, A. M. Jorge, and A. Jatowt, "Survey of temporal information retrieval and related applications," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, article no. 15, 2015.
- [83] K. Lee, Y. Artzi, J. Dodge, and L. Zettlemoyer, "Context-dependent semantic parsing for time expressions," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, MD, 2014, pp. 1437-1447.
- [84] H. Ji, T. Cassidy, Q. Li, and S. Tamang, "Tackling representation, annotation and classification challenges for temporal knowledge base population," *Knowledge and Information Systems*, vol. 41, no. 3, pp. 611-646, 2014.
- [85] T. Cassidy, "Temporal information extraction and knowledge base population," Ph.D. dissertation, City University of New York, NY, 2014.
- [86] Y. S. Jeong, Z. M. Kim, H. W. Do, C. G. Lim, and H. J. Choi, "Temporal information extraction from Korean texts," in *Proceedings of the 19th Conference on Computational Natural Language Learning*, Beijing, China, 2015, pp. 279-288.
- [87] Y. S. Jeong and H. J. Choi, "Language independent feature extractor," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, Austin, TX, 2015, pp. 4170-4171.
- [88] S. Bethard, G. Savova, W. T. Chen, L. Derczynski, J. Pustejovsky, and M. Verhagen, "Semeval-2016 task 12: clinical TempEval," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval)*, San Diego, CA, 2016, pp. 1052-1062.
- [89] S. MacAvaney, A. Cohan, and N. Goharian, "GUIR at SemEval-2017 Task 12: a framework for cross-domain clinical temporal information extraction," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval)*, Vancouver, Canada, 2017, pp. 1024-1029.
- [90] P. R. Sarath, R. Manikandan, and Y. Niwa, "Hitachi at SemEval-2017 Task 12: system for temporal information extraction from clinical notes," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval)*, Vancouver, Canada, 2017, pp. 1005-1009.

- [91] A. Leeuwenberg and M. F. Moens, "KULeuven-LIIR at SemEval-2017 Task 12: cross-domain temporal information extraction from clinical records," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval)*, Vancouver, Canada, 2017, pp. 1030-1034.
- [92] J. Tourille, O. Ferret, X. Tannier, and A. Neveol, "LIMSI-COT at SemEval-2017 Task 12: neural architecture for temporal information extraction from clinical narratives," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval)*, 2017, pp. 597-602.
- [93] E. Laparra, D. Xu, A. Elsayed, S. Bethard, and M. Palmer, "SemEval 2018 Task 6: parsing time normalizations," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, New Orleans, LA, 2018, pp. 88-96.
- [94] S. Bethard and J. Parker, "A semantically compositional annotation scheme for time normalization," in *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*, Portoroz, Slovenia, 2016, pp. 3779-3786.
- [95] A. Olex, L. Maffey, N. Morgan, and B. McInnes, "Chrono at SemEval-2018 Task 6: a system for normalizing temporal expressions," in *Proceedings of the 12th International Workshop on Semantic Evaluation*, New Orleans, LA, 2018, pp. 97-101.
- [96] J. Pustejovsky and A. Stubbs, *Natural Language Annotation for Machine Learning: A Guide to Corpus-Building for Applications*. Sebastopol, CA: O'Reilly Media, 2012.



Chae-Gyun Lim <https://orcid.org/0000-0002-4534-4005>

He is currently a PhD candidate in the School of Computing at Korea Advanced Institute of Science and Technology (KAIST), Korea. In 2011, he received a B.S. in Medical Computer Science from Eulji University, Korea. Between 2011 and 2013, he worked as a research assistant in the Department of Computer Science at KAIST, Korea, and in 2015, a M.Sc. in the Department of Computer Engineering at Kyung Hee University, Korea. His research interests include temporal information extraction, topic modeling, big data analysis and bioinformatics.



Young-Seob Jeong <https://orcid.org/0000-0002-9441-2940>

He received a B.S. in Computer Science from Hanyang University, Korea, in 2012, an M.Sc. in Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Korea, and in 2016, a Ph.D. in School of Computing from KAIST, Korea. He joined the faculty of the Department of Big Data Engineering at Soonchunhyang University, Asan, Korea, in 2017. His current research topics are text mining, information extraction, action recognition, and dialog systems, where his favorite techniques are topic modeling and deep learning.



Ho-Jin Choi <https://orcid.org/0000-0002-3398-9543>

He is currently an associate professor in the School of Computing at Korea Advanced Institute of Science and Technology (KAIST). In 1982, he received a B.S. in Computer Engineering from Seoul National University, Rep. of Korea. In 1985, he obtained an M.Sc. in Computing Software and Systems Design from Newcastle University, UK, and in 1995, a Ph.D. in Artificial Intelligence from Imperial College, London, UK. Currently, he serves as a member of the board of directors for the Software Engineering Society of Korea, the Computational Intelligence Society of Korea, and the Korean Society of Medical Informatics. His current research interests include artificial intelligence, data mining, software engineering, and biomedical informatics.