Semantic Patterns to Structure TimeFrames in Text

Nour Matta
IT and Digital Society Laboratory
University of Technology of Troyes
Troyes, France
email: Nour.matta@utt.fr

Nada Matta
IT and Digital Society Laboratory
University of Technology of Troyes
Troyes, France
email: Nada.matta@utt.fr

Nicolas Declercq
Namkin
Troyes, France
email: n.declercq@namkin.fr

Agata Marcante
Research & Development
Namkin
Troyes, France
email: a.marcante@namkin.fr

***Abstract*— Event ordering is a very important task in the event extraction field since any analysis of the causality and impacts of a specific action or a change requires consideration of temporality and ordering. Many pattern-based approaches or machine learning approaches work on identifying the events in the text and creating relationships between them. In this paper, we present a novel approach based on timeframes, that will enable distinction between multiple timeframes in a text, when available, and grouping events within these timeframes.**

Keywords-Timeframe; Event Extraction; Event Ordering; Natural Language Processing.

I. INTRODUCTION

Event extraction is one of the most important tasks of Information Extraction through Natural Language Processing [25]. It enables the extraction of events in text and aims to identify the different participants and attributes of the extracted events. Some examples of the extracted information can be the cause, place, time, means, or goal, that can be identified through dependency analysis [28]. Moreover, evaluating the influence of a particular event or a specific action requires an account of temporality [27]. In the traditional event extraction, available approaches are very performant when it comes to the analysis of single sentences. Some approaches can support complex sentences. But even though models aim to extract events from a "text" and create temporal relations between them, the performance lacks and soon the extracted information easily becomes unreadable or, from a temporal relation point of view, inaccurate [8]. Furthermore, when focusing on the temporality event extraction, many approaches focused on ordering the multiple events mentioned one by one and creating relationships [12], [26] without considering the fact that multiple 'processes' can be part of a preparatory stage of a single event. In this paper, we introduce the use of timeframes, an approach used for the time analysis in different domains, to improve the temporal relation made between events in a text.

It is important to note that within the same text, multiple timeframes can be identified, and multiple time references can be used. A small example would be a news report about a company announcing the launching of a new product. We have the time when the announcement was made, the timeframe within the announcement (such as the date of the launching), and the time of the publication of the news. Another example would be in a narrative text in which the author talks about multiple events while going back and forth in time. Our main goal is to identify the events in a text, create temporal relations between them and identify the different timeframes if there are multiple ones. We aim to assign each event to its timeframe enabling improved readability of the extracted event, their temporal relation, and finally their interpretation.

In Section 2, we will start by defining what timeframes are and how they are used for time analysis. We will also present the different conceptualizations of events in linguistics. In Section 3, the related work, we will go through the different event extraction approaches before presenting the timeframe approach along with part of the semantic pattern identified.

II. TIMEFRAMES AND EVENTS

This section is divided into two main parts: the timeframes and the events. For the timeframe, we will go through the analysis of temporality in fields other than text mining and show how those conceptualizations can be helpful in the analysis of temporality in text. As for the events, we will go through their definition in the event extraction field and how it is viewed from a linguistic point of view.

A. Timeframes

A timeframe is a certain period of time in which an event should happen or has already happened [3]. This leads us to question the meaning of time. In philosophy [19], the platonist understanding of time is segregated from the relationist definition. Platonists picture time as an "empty container" of events that exists regardless of whether anything is placed in it. In this perspective, platonists consider that it is possible that changes in the universe can cease to exist for a certain period. On the other hand, relationists view time as a set of events and the temporal relationship between these events. While dealing with event extraction and their temporal relationship, the relationist understanding is used.

The study of the temporal relation between actions, events, states, and their influences is applicable in different domains other than event extraction. In their study on temporality in video games, Zagal et al. [27] distinguished multiple types of games: the ones with game time being equivalent to the real-world time, the ones in which action can speed up or skip time, the ones where specific action triggers events of a specific duration, and finally the ones where certain events occur without affecting the game time as if time had stopped. To analyze the temporality for each game type, Zagal et al. [27] defined timeframes, creating relations between those timeframes and between events within the timeframe and coordinating them. Reflecting on that approach, from a textual perspective, the authors also set the duration to specific events as shown in "1)", which can make flashbacks "2)" and flash-forwards "3)". They can also skip time "4)" and even focus on a specific event or describe elements making the time indirectly stop "5)".

1) *John ran for an hour.*
2) *Henry was looking at the photo. He took it a few years back when he was in New York.*
3) *John is preparing his luggage; he will be leaving in the morning.*
4) *Five years later, Henry went back to New York.*
5) *John looked through the window for a few seconds. It was a rainy day; people were walking while holding their umbrellas. He went to his desk.*

Distinguishing the different timeframes and specifying the events that happened in each frame enables the focus on specific events based on their occurrence time and aims to improve coordination between multiple timeframes in the text. However, in the event extraction field, events are ordered one by one without having a more global representation. Some of the concepts that must be considered while dealing with temporality are the duration, the time point, calendar, narrative time, timeline, countdown, and temporal relation [3], [27]. Each of these concepts plays a specific role in the pattern and the extracted knowledge. The use of timeframe also enables the consideration of the release date of the text as a timeframe on its own in order to improve topic tracking and event follow-up.

B. Events

In the event extraction field and the event-based decision systems, events are usually defined as happenings or changes that occurred in a specific interval of time. They can be associated with the change of states (canceled, ongoing, recently done, past or future plans) and can have multiple occurrences [17], [25].

Other than Natural Language Processing, linguists also worked on defining what an event is, distinguishing it from a state, and partitioning it onto atomic and extended events. Using the tense of the verb, the duration, and time reference along with temporal connectors, some set of rules and patterns are proposed. Vendler [23] was one of the first to work on defining the concept of event in linguistics, while working on verbs and tenses, he first identified the tense as the location of a happening in the time (past, present, or future) and its aspect which refers to the state of an event (completed, ongoing or interrupted). Later on, he defined "Eventualities" [23] as a concept that groups the state and non-state. Some particularities of each group were identified:

6) *Jack was ill on Sunday.*
7) *Jack wrote a letter on Sunday.*

"6)" is an example of a state, in which we cannot determine if the state "ill" started before or during the "Sunday" and if is stopped during "Sunday" or after. While the non-state "wrote" started on ended Sunday. And comparing the duration of "was ill" and wrote, we can presume that "wrote" has a shorter duration than "was ill".

Their conceptualization of eventualities goes as follows: non-state was divided into Activities and Events; activity refers to actions that had a duration but with no endpoint or consequent state while events have a quantification or an ending result:

8) *Alex ran.*
9) *Alex ran to the store.*
10) *Alex ran a mile.*

"8)" is considered an activity while "9)" and "10)" are events. Events are then distinguished [24]. Where an accomplishment is considered to have a duration and accept the progressive (continuous tense) while achievement is strange in progressiveness. It is important to note that Kamp highlighted the ambiguity between those concepts, starting with the very first division between distinguishing a state and a nonstate.

Using the conceptualization made by Vendler, Moens et al. [17] defined another conceptualization. Eventualities are divided into States and Events. And they considered two dimensions for distinguishing events: the duration and the consequence. For the duration, they considered events as Atomic or Extended. Extended Events have a notion of duration. For the consequence, they started by defining the term "culmination" as an event that has a consequence, a change of state. A "nucleus", as shown in Figure 1, is the combination of a preparatory process of a culmination, the culmination, and the consequent state. If we consider the example "9)", "Alex ran to the store", we can regard it as a culmination, in which running is a preparatory process to the culmination of arriving at the store. The consequent state is "being in the store".

Figure 1. Moens et al. nucleus definition [17].

 ISBN: 978-1-61208-977-5

Figure 2 presents the 4 subcategories of events. An Atomic Event with no consequence is considered as a point, for instance, "He hiccupped". A culmination is an atomic event with a consequence [17]. An extended event is a process and is considered a culminated process if it has a consequence. It is important to highlight that many elements were used to distinguish the different categories of events. This work and the pattern identified in it are essential for our approach especially the use of a nucleus. When using the timeframe approach, identifying the culminations in a text and all the processes, the preparatory stage, and the consequent state are one of our goals. The slice difference in our approach is that we intend to associate different events on the nucleus timeline.

	Atomic Event	Extended Event
- Consequence	Point	Process
+ Consequence	Culmination	Culminated Process

Figure 2. Moens et al. event conceptualization [17].

The pattern identified by Moens et al. will be used and associated with other patterns to enable the representation of events extracted for each timeframe. When trying to identify events and states, the use of adverbs, the tense of the verb, and the use of semantic dependencies, such as the verbs' objects, were used for identifying the categories. The same verb can be considered a point, a process, a culmination of a culminated process depending on its use.

It is important to note that in linguistics, verbs tend to be classified as states and events. Adjectives are considered states of the elements they describe. In the event extraction field, nouns are also identified as events depending on their context, for example:

11) *Two years after his graduation, John moved to New York.*

In this sentence, the noun "graduation" is considered as an event. In order to manage these types of events, a pattern concerning nouns was added. If a noun is a temporal reference, in this case, "after graduation", then this noun is an event. Several studies define principles to identify and extract events along with their arguments; we summarized them in the related work.

III. Related Work

This section will be partitioned as follows: we will start by going through event extraction techniques and more importantly Event Ordering. Then we will go through different research works that addressed temporality aspects in the temporality recognition techniques.

A. *Event Extraction and Even Ordering*

There are two main event extraction types: the closed-domain and the open-domain [26]. The closed-domain event extraction refers to the detection of specific events of interest, for example, the merger of two companies. In this case, the information related to the event is predefined. This approach is usually used for event mention detection, event trigger, and event argument [12]. Some classify the role of each argument that is specific to the event of interest. As for the open domain, the search of events is not bounded to specific events and aims to detect all sorts of events within a text and later on cluster texts based on similar events detected [8]. This approach is usually used for story segmentation, first story detection, topic detection, etc. In this paper, we focus on the open domain event extraction since applying the approach to a specific domain is out of our scope. In order to extract or detect events, multiple approaches are available from pattern matching to machine learning [13], deep learning [10], semi-supervised learning [7], and unsupervised learning [26]. This depends on the type of events of interest, if there are pre-trained models, and the purposes of the extraction.

Event Ordering is a branch in event extraction that focuses on extracting events and creating temporal relations between them [15]. Multiple annotated datasets are available to train models, the most popular being the TimeBank-Dense corpus [4]. This corpus has three types of relations, the intra-sentence, the cross sentence, and the document creation time. Note that one approach was considered as a "context-aware" model for using all three types of relations [16]. Temporal label dependencies and constraints are used to improve relations between events [1]. Some worked on the linguistic and syntactic rules, such as Leeuwenberg et al. [11] or Laokulrat et al. [10]. Please note that, in this paper, we will be using the defined constraint, rules, and linguistic features to enrich our approach. The relation between a timeframe and events will be added with consideration of culminations and constraints on culminations.

B. *Temporality Recognition Techniques*

In the Question Answering field, temporal analysis is a must for determining if an answer to a question will change throughout time or not. In their work, Pal et al. [20] identified multiple classes of information temporality: short duration, medium duration, long duration, and permanent. They tried classifying the question/answer under those categories but ended up grouping the short term and medium term together and long term and permanent together. It enabled distinguishing between "Who won the competition X in 2022?" and "Who won the last competition?". One of the questions will have permanent information and the other will change throughout the years. It is important to consider these types of classifications to identify information that is true regardless of the timeframe of the text and relations that are relative to the timeframe of the text. Recent work focuses on identifying the attention in complex questions and the use of multiple sentences that contain the answer [5]. Note that in their work Kwiatkowski et al. [9] mentioned descriptive sentences or informative sentences, in which information is given without a particular event being mentioned.

Temporality plays a very important part in social science and social discourse analysis [6]. Coordination between different events from multiple resources is also used when clustering news and following up on events. Sources vary between news and social media posts, such as tweets [18]. It is also essential to consider time relations when analyzing the influence of social media and the media in general on social events, such as protests and violence and study the sentiments behind it [22].

The question answering field provided a very important aspect to consider when extracting events and information. Completed Events and states with a specific date tend to be permanent information while unfinished events and events with reference to the text temporality tend to be true in a specific timeframe. Coordination of events between multiple texts will be considered in our approach and will be based on the timeframe concept. Our approach introduces the use of multiple types of timeframes and how to extract them. We will be using several models and patterns already provided in order to optimize the model's performance.

IV. TIMEFRAME APPROACH

For the extraction of timeframes that will be used to improve the temporal relation analysis between events, we identified three types of timeframes, (1) the Publication Timeframe, (2) the Narrative Timeframe, and (3) the Spoken Timeframe. Those timeframes were inspired by the identified timeframes for temporal analysis in video games with adaptation to the text constraints [27]. The Publication Timeframe reflects the publication date or year of the analyzed text. The Narrative Timeframe is the timeframe of the events happening in the text; we may find multiple Narrative Timeframes in a single document. Finally, the Spoken Timeframe is a particular type of timeframe that may not always appear in a text. It is used when an announcement, a speech, or a dialogue is present. The events and information that are mentioned in that context will be analyzed in their own timeframe in order to reduce event relationship complexity. The timeframe will consist of two main parts: (1) the text belonging to the timeframe, and (2) the extracted information related to it.

A. *Publication Timeframe Extraction*

All text document have by default a Publication Timeframe and a Narrative Timeframe. To identify the publication date, the type of text affects the extraction. If a post on social media is being analyzed then, the date is usually available as metadata to the text. When dealing with online news, most publishers put the date at the beginning of the text. Considering the presence of the title, we will check the first three sentences, for the presence of dates using Named Entity Recognition. If no dates were found, the last sentence will be checked. In case a sentence was identified as the publication date, it will be extracted from the document in order to avoid confusion with the rest of the text. The date will be set in the information field of the timeframe. Figure 3 provides the Publication Timeframe extraction function. It takes two elements as input: a text, and the patterns that identify the publication date. The returned list contains two elements: the Publication Timeframe and the text. The text is returned since it is modified in case the pattern was found in a sentence.

```
Extraction_Publication(Single_Text, pub_pattern):
    sent = split_sentences(single_Text)
    tf_pub = [[],[]]
    to_check = [sent[0],sent[1],sent[2],sent[-1]]
    for element in to_check:
        identified = check_pattern(pub_pattern,element)
        if identified:
            timeframe_pub = [[date],[element]]
            remove element from Single_Text
            return [timeframe_pub , Single_Text]
    return [[], Single_Text]
```

Figure 3. Publication Timeframe Extraction.

Table 1 provides some of the patterns used to identify the Publication Timeframe. Please note that for the first two patterns, their presence in the sentence is enough while for the last two, they must be alone in the sentence to be considered a sign of Publication Timeframe.

TABLE I. SOME OF THE PATTERN USED TO DETECT PUBLICATION TIMEFRAMES

pub_patterns
'Updated' + <date>
'Published' + <date>
<number> + [hours, days, months, years] + 'ago'
<date>

B. *Spoken Timeframe Extraction*

```
Extraction_Spoken(Single_Text, say_pattern):
    list_sentence = split_in_sentences(single_Text)
    tf_speech_element = [[],[]]
    tfs_speech =[]
    id = 0
    before = false
    for sentence in list_sentence:
        identified = check say_pattern in sentence
        if identified and not before:
            before = true
            tf_speech_element = [id,[sentence]]
            replace(sentence,Single_Text,"tf_speech_"+id)
        else:
            if identified and before:
                add sentence to tf_speech_element[2]
                remove sentence from Single_Text
            else:
                if before:
                    before = false
                    add tf_speech_element to tfs_speech
                    id = id + 1
                    tfs_speech = [[],[]]
    return [tfs_speech , Single_Text]
```

Figure 4. Spoken Timeframe Extraction.

This timeframe will be treated before the Narrative Time. The search for verbs that reflect speaking and punctuation that are proper to dialogue will be the main task. If nothing is identified, we skip to the next stage, else a spoken frame will be created. If the "spoken" elements are all available in successive sentences, they will all be extracted and set in a single Spoken Timeframe. If multiple sentences have 'spoken' elements but are not successive, a Spoken

Timeframe should be created for each nonconsecutive part. But in order to enable relations between the Narrative Timeframe and the Spoken Timeframe, identification will be assigned to each extracted Spoken Timeframe, and the extracted sentences will be replaced by the Spoken Timeframe Identification. If any dates are mentioned, they can be added to the information field of the timeframe. Note the tense in the Spoken Timeframe reflects a relationship between the Spoken and Narrative Timeframe it belongs to, so if a unique tense is identified, a relation between the Spoken and the Narrative Timeframe will be identified. For example, if future tense is identified in the "spoken" element, then the relationship will most probably be "after". To keep track of this relationship, the relation if available will be added with the timeframe identification.

Figure 4 provides the Spoken Timeframe extraction function. It takes two elements as input: a text, and the patterns that identify the speaking patterns. The returned list contains two elements: the Spoken Timeframe list and the text. The Spoken Timeframe list contains all the Spoken Timeframes identified in the text. Each one contains an identification that distinguishes different segments in which the patterns were identified along with the sentences. Note that if consecutive sentences contain the patterns, then they will be grouped in the same timeframe. The return text is the remaining text with the identifications of the Spoken Timeframes.

We used a single pattern to identify the presence of a direct speech in a sentence. First, some direct speech may contain multiple sentences between quotation mark which reduced the accuracy of the dependency parsing. This is why, during the pattern matching phase, any text between quotation was replaced by "" and we analyzed the dependencies of the quotations. If the quotation mark is the object of the verb in the sentence, then we consider that the current sentence belongs to a Spoken Timeframe.

C. Narrative Timeframe Extraction

The starting point of the Narrative Timeframe is having an empty information field and the whole text inside of it. The purpose of using multiple timeframes is to distinguish between current time in a text and in case a flashback is mentioned, or flash-forward is mentioned, the information should be treated accordingly. Using the VerbNet parser [2], we detect any temporal relation. We associate a change in the timeframe when the relationship is not related to a specific event. For example, "before going to bed" is related to the event "go to bed" while "a few years ago" is a temporal relation with the current timeframe. We also consider "later that day" or "later that year" elements within the same timeframe.

In this section, we will present the elements that trigger the creation of a new Narrative Timeframe. The temporal relationship elements that will create a new Narrative Timeframe are: "a few years later", "(number) years later". The same goes for "months" and "days" instead of "years" and "ago" instead of "later". Dates are relatively important; if a date is mentioned, it will be assigned as information about the timeframe. If no dates are mentioned, temporal relations that start with 'this' for example, 'this year', 'this month', 'today', will be considered as time information of the timeframe. If multiple dates are separately mentioned, each will be assigned a timeframe.

```
Extraction_Narrative(Single_Text, narrative_pattern, tense_pattens):
    list_sentence = split_in_sentences(single_Text)
    tf_nar_element = ["tf_nar_0",[],[]]
    tfs_narrative = []
    id = 0
    tocheck = 0
    for sentence in list_sentence:
        identified = check narrative_pattern in sentence
        if tocheck == 0:
            if not identified :
                add sentence to tf_nar_element[1]
            else:
                dominant_tense = check_tense(tf_nar_element)
                id=id+1
                add "tf_nar_"+id to tf_nar_element[1]
                add tf_nar_element to tfs_narrative
                sentence_tense = check_tense(sentence,tense_pattens)
                tf_nar_element = ["tf_nar_"+id,[sentence],[sentence_tense]]
                if (dominant_tense != sentence_tense):
                    tocheck = 1
        else:
            if not identified :
                sentence_tense = check_tense(sentence)
                if sentence_tense == tf_nar_element[2]:
                    add sentence to tf_nar_element[1]
                else:
                    add tf_nar_element to tfs_narrative
                    id=id+1
                    sentence_tense = check_tense(sentence)
                    tf_nar_element = ["tf_nar_"+id,[sentence],[sentence_tense]]
                    tocheck = 0
    add tf_nar_element to tfs_narrative
    return tfs narrative
```

Figure 5. Narrative Timeframe Extraction.

Figure 5 provides the algorithm used for the Narrative Timeframes extraction. It takes as input the text, the patterns that identify the existence of a new timeframe, and the patterns that check the tense of a verb. The patterns that check the tense of the verbs are based on the part-of-speech tagging, dependencies, and the lemmatization of the verb. The lemmatization is the original form of a word without conjugation. We use it only to detect the verbs 'be' and "have". Those elements are provided by Natural Language Processing tools such as Spacy [21]. For the part-of-speech tags of interest, we used:

- "VB" is assigned to the verbs base form
- "VBD" is assigned to verbs in the past tense
- "VBG" is the gerund (a verb that ends with 'ing')
- "VBN" assigned to the verb in past participle form
- "VBP" is assigned to the verbs in non-third person singular present form
- "VBZ" is assigned to the verbs in the third person singular present form

We considered the 12 principal tenses, and Table 2 provides some of the tenses and their respective patterns. We grouped the 12 verb tenses in the respective 5 tense categories: past anterior, past, present, future, future anterior [23]. For example, present continuous and present simple will both be present while present perfect, past simple, and past continuous will be considered as past. Based on the verb tense, the function check_tense will return the category of the verb tense identified.

TABLE II. SOME OF THE PATTERN USED TO DISTINGUISH VERB TENSE

Verb Tense	tense_patterns
Present Simple	pos = "VBZ" or pos = "VBP"
Present Continuous	verb with pos="VBG" and has_child = {dep= "AUX", pos = "VBZ" or "VBP", lemma="be"}
Past Simple	verb with pos="VBD"
Past Continuous	verb with pos="VBG" and has_child = {dep= "AUX", pos = "VBD", lemma="be"}
Future Simple	verb with pos="VB" and has_child = {dep= "AUX", pos = "MD", lemma="be"}

As for the patterns that identify the presence of a new Narrative Timeframe, Table 3 presents some of them.

TABLE III. SOME OF THE PATTERNS THAT IDENTIFY NARRATIVE TIMEFRAMES

Narrative_Patterns
A few ['years', 'months', 'days'] ['later', 'ago', 'back']
['earlier', 'later'] ['this', 'that'] ['years', 'months', 'days']
In <date>
['starting', 'from', 'starting from'] <date>
<number> ['years', 'months', 'days'] ['later', 'ago', 'back']

The algorithm goes as follows: an empty Narrative Timeframe is initialized. We go through all the sentences and we check the presence of a pattern. If no pattern is identified, we add the sentence to the timeframe. If patterns that trigger the creation of a new Narrative Timeframe are identified, we generate an identification to the new timeframe and we add to the previous timeframe to keep track of their connection. We then check the tense of the previous timeframe and the tense of the new one and we save the current timeframe element in the list of Narrative Timeframes. If the tenses are similar, we just add the sentence to the new timeframe. If the tenses are similar, there is a high risk that the author switches back to the previous timeframe. In that case, for the upcoming sentences, we keep track of any changes in the tenses. This is the only case in which the change of tense will trigger a change in the Narrative Timeframe. In future work, just like the event ordering approaches, a change in tense will trigger relations between events. Examples (12) and (13) clarify the need for this process:

12) *John is thinking about his life in New York. A few years ago, he had to move out because of his parents' job. He misses his friends dearly.*
13) *Alice graduated with a master's degree. A few months later, she found a job in an international company. She was finally able to move out.*

In 12), the change of the tense use can simulate a go back to the previous timeframe or just a need to change timeframes. In our current algorithm, we will just separate the three timeframes and handle the relationships between different timeframes in future works. As for 13), the continuity in the tense simulates just a skip in time with no need for further tense monitoring of verb tenses. In the next section, we will be evaluating our approach.

V. EXAMPLE OF APPLICATIONS

In this section, we will present an output for each of the three algorithms provided above. Please note that for evaluating the output of the timeframe approach we used two types of resources, posts from LinkedIn and news scraped from multiple online news sites. For the LinkedIn posts, we focused on companies' accounts. We distinguish two types of posts, the short ones, and the long ones. For the long post, we selected 20 posts that contained multiple Narrative Timeframes or the need for Spoken Timeframes. As for news data, the scraping was made from three different sites in three different fields: politics, industry, and global. We took 10 news from each online source. The websites used were CNN (edition.cnn.com), BBC (bbc.com), and GlobalNews (globalnews.ca). We used this small amount of data in order to compare the output with the expected output manually since the approach is still in its early stages.

A. Publication Timeframe Extraction Results

The publishing timeframe extraction was not needed since the data was extracted "structured", with a distinction between the text and the date of publication. The Publication Timeframe was only applied to news data. For each site, we started by testing the performance of the Publication Timeframe since the scraper used is not customized for each website. We were able to identify the Publication Timeframe. Please note that a cleaning phase is necessary before applying the approach. Figure 6 presents one of the outputs of the algorithms. The figure provides part of the extracted text from a news site with the sentence with the pattern of interest highlighted in the input of the algorithm. We can notice the Publication Timeframe along with the rest of the text from which we removed the sentence with the pattern.

Figure 6. Output of Publication Timeframe Extraction Algorithm.

B. Spoken Timeframe Extraction Results

The Spoken Timeframe Extraction was applied to both data sets, the LinkedIn data, and the news data. Figure 7 provides the output of one of the provided data. Please note that for better visualization, long paragraphs with no pattern were replaced by '...' in the figure.

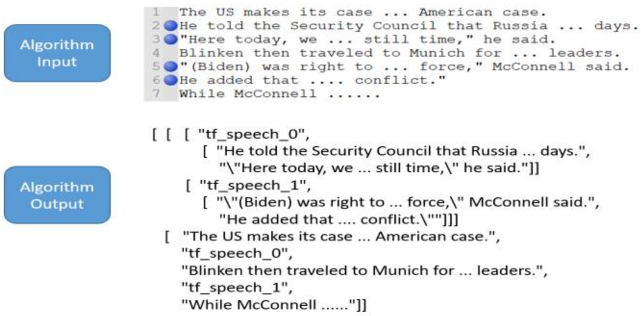

Figure 7. Output of Spoken Timeframe Extraction Algorithm.

In the example provided in Figure 7, we identified 4 sentences with patterns. They were distributed into 2 groups of consecutive sentences with patterns. The sentences in the example input were marked by blue dots next to them. In the output, we can notice that the algorithm provided a list with two elements in it, a Spoken Timeframe list and a list of the remaining sentences. The Spoken Timeframe list had 2 Spoken Timeframes in it, each having identification and consecutive sentences with patterns. As for the remaining text list, we notice that the extracted sentences were indeed replaced by their respective timeframe identification.

C. Narrative Timeframe Extraction Results

Finally, for the Narrative Timeframe, we used a text that had 2 Spoken Timeframes already identified in it. We also added a blue dot next to the sentences with the patterns identified. We can notice in the output the Narrative Timeframe list returned by our algorithm in Figure 8. It contains the three expected timeframes having their respective identification, the sentences ordered that belong to the timeframe, and the tense of the last sentence to enable comparisons.

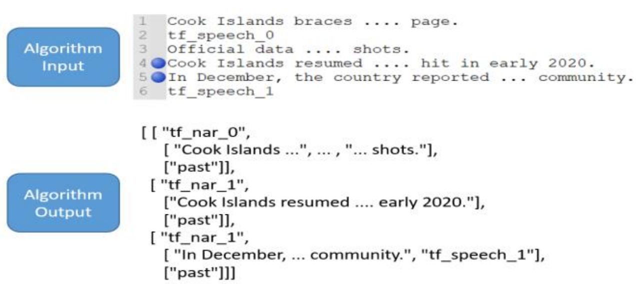

Figure 8. Output of Spoken Timeframe Extraction Algorithm.

D. Results Analysis

Out of the 20 LinkedIn posts, the Spoken Timeframe was only available in one post due to the nature of posts on social media. But the Narrative Timeframe was performant, and we were able to segment the text as needed. We noticed one case of none-identified timeframes due to an unavailable pattern that was added later on to the model. As for the news data, the Spoken Timeframes were highly identified in all fields due to the announcement or relay of speech of people. Out of the 30 news, only eleven showed multiple Narrative Timeframes. These timeframes were mostly used in news data when a follow-up on a story happens. In most cases, recent events are stated, before mentioning what has happened 'earlier' regarding the same story. For events that are anticipated to happen, the timeframes were not separated and will be considered in the future as the relation between events and not timeframes.

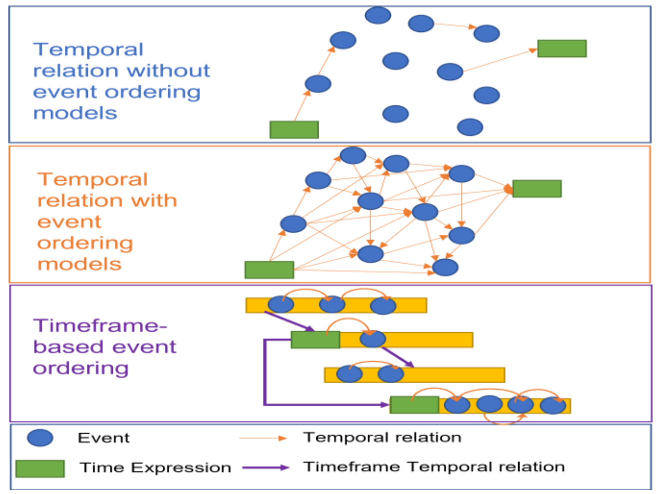

Figure 9. Temporal relations representation of multiple approaches.

Figure 9 shows the difference in the representation of temporal relations extract using multiple approaches. The first output shows a representation of an output without using an event ordering model. We can notice that hardly 4 events are connected and only 2 events are related to a time expression. The second presents the output of the same event extraction but after applying an event ordering model. This time most events are connected but the interpretation and the usability are complex. Finally, the third provides our desired representation using the temporal timeframes. This approach grouped events that occurred in the same period of time and limited the relation extraction between events from different timeframe. In this article we proposed the different type of timeframes and how to extract them without extracting the relationships between the timeframes.

VI. CONCLUSION

Finally, event extraction is an essential task in the NLP field. It enables the use of text data in order to build decision-making systems and for event monitoring. In this paper, we highlighted the need for timeframes to improve event ordering in the event extraction field. Three types of timeframes were presented: the publication, the narrative, and the Spoken Timeframe. Publication Timeframes will be used for multiple text analysis as a temporal indicator of the text. Narrative Timeframes enable the distinguishing of multiple periods of time used in a text, notably when a flashback or a flash-forward occurs. Finally, the Spoken Timeframe enables the distinction between the Narrative

 ISBN: 978-1-61208-977-5

Timeframes and the timeframe of "spoken" elements in a text, such as announcements or dialogs. We set a few patterns for the identification and extraction of the different timeframes. In future work, we will provide relation extraction methods for the timeframes and the events of each timeframe. We intend to evaluate the performance on longer texts and a larger number. We will also be distinguishing the multiple classes of events: point, process, culmination, and culminated process in order to identify states available in timeframes. This work will complete our study on detection and representation context from text [14].

References

[1] S. Bethard, "ClearTK-TimeML: A minimalist approach to TempEval 2013," in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Atlanta, Georgia, USA, Jun. 2013, pp. 10–14.

[2] S. W. Brown, J. Bonn, J. Gung, A. Zaenen, J. Pustejovsky, and M. Palmer, "VerbNet Representations: Subevent Semantics for Transfer Verbs", in *Proceedings of the First International Workshop on Designing Meaning Representations*, Florence, Italy, Aug. 2019, pp. 154–163.

[3] M. J. Buehner and J. May, "Knowledge mediates the timeframe of covariation assessment in human causal induction," *Thinking & Reasoning*, vol. 8, no. 4, pp. 269–295, Nov. 2002, doi: 10.1080/13546780244000060.

[4] N. Chambers, "NavyTime: Event and Time Ordering from Raw Text," in Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), Atlanta, Georgia, USA, Jun. 2013, pp. 73–77. Accessed: Feb. 07, 2022. [Online]. Available: https://aclanthology.org/S13-2012

[5] X. Duan *et al.*, "Temporality-enhanced knowledgememory network for factoid question answering," *Frontiers Inf Technol Electronic Eng*, vol. 19, no. 1, Jan. 2018, pp. 104–115.

[6] J. Hamann and L. Suckert, "Temporality in Discourse: Methodological Challenges and a Suggestion for a Quantified Qualitative Approach," *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, vol. Vol 19, p. No 2 (2018), Mar. 2018, doi: 10.17169/FQS-19.2.2954.

[7] R. Han, Q. Ning, and N. Peng, "Joint Event and Temporal Relation Extraction with Shared Representations and Structured Prediction," *arXiv:1909.05360 [cs]*, Sep. 2020.

[8] F. Hogenboom, F. Frasincar, U. Kaymak, F. de Jong, and E. Caron, "A Survey of event extraction methods from text for decision support systems," *Decision Support Systems*, vol. 85, pp. 12–22, May 2016, doi: 10.1016/j.dss.2016.02.006.

[9] T. Kwiatkowski *et al.*, "Natural Questions: A Benchmark for Question Answering Research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 453–466, Aug. 2019, doi: 10.1162/tacl_a_00276.

[10] N. Laokulrat, M. Miwa, Y. Tsuruoka, and T. Chikayama, "UTTime: Temporal Relation Classification using Deep Syntactic Features," in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Atlanta, Georgia, USA, Jun. 2013, pp. 88–92.

[11] A. Leeuwenberg and M.-F. Moens, "Structured Learning for Temporal Relation Extraction from Clinical Records," in Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Valencia, Spain, Apr. 2017, pp. 1150–1158. Accessed: Feb. 09, 2022. [Online]. Available: https://aclanthology.org/E17-1108

[12] Q. Li *et al.*, "Reinforcement Learning-Based Dialogue Guided Event Extraction to Exploit Argument Relations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 520–533, 2022.

[13] Y. Lu *et al.*, "Text2Event: Controllable Sequence-to-Structure Generation for End-to-end Event Extraction," *arXiv:2106.09232 [cs]*, Jun. 2021.

[14] No. Matta, N. Matta, E. Giret, and N. Declercq, "Enhancing Textual Knowledge Discovery using a Context-Awareness Approach", in *The 8th Annual Conf. on Computational Science & Computational Intelligence (CSCI'21)*, December 15-17, 2021, USA, in press.

[15] B. McDowell, N. Chambers, A. Ororbia II, and D. Reitter, "Event Ordering with a Generalized Model for Sieve Prediction Ranking," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Taipei, Taiwan, Nov. 2017, pp. 843–853.

[16] Y. Meng and A. Rumshisky, "Context-Aware Neural Model for Temporal Information Extraction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia, Jul. 2018, pp. 527–536. doi: 10.18653/v1/P18-1049.

[17] M. Moens and M. Steedman, "Temporal Ontology and Temporal Reference," *Computational Linguistics*, vol. 14, no. 2, pp. 15–28, 1988.

[18] V. Moutinho, João Cordeiro, and P. Brazdil, "Association and Temporality between News and Tweets," pp. 500–507, Sep. 2019, doi: 10.5220/0008362105000507.

[19] W. H. Newton-Smith, "The Structure of Time," *Routledge & CRC Press*, Jun. 01, 2020.

[20] A. Pal, J. Margatan, and J. Konstan, "Question temporality: identification and uses," in *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, New York, NY, USA, Feb. 2012, pp. 257–260. doi: 10.1145/2145204.2145246.

[21] E. Partalidou, E. Spyromitros-Xioufis, S. Doropoulos, S. Vologiannidis, and K. I. Diamantaras, "Design and implementation of an open source Greek POS Tagger and Entity Recognizer using spaCy", in *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, Oct. 2019, pp. 337–341.

[22] T. Poell, "Social media, temporality, and the legitimacy of protest," *Social Movement Studies*, vol. 19, no. 5–6, pp. 609–624, Nov. 2020, doi: 10.1080/14742837.2019.1605287.

[23] Z. Vendler, "Verbs and Times," *The Philosophical Review*, vol. 66, no. 2, pp. 143–160, 1957, doi: 10.2307/2182371.

[24] Z. Vendler, *Linguistics in Philosophy*. Cornell University Press, 2019. doi: 10.7591/9781501743726.

[25] X. Wu, T. Wang, Y. Fan, and F. Yu, "Chinese Event Extraction via Graph Attention Network," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 21, no. 4, p. 71:1-71:12, Jan. 2022, doi: 10.1145/3494533.

[26] W. Xiang and B. Wang, "A Survey of Event Extraction From Text," *IEEE Access*, vol. 7, pp. 173111–173137, 2019

[27] J. Zagal and M. Mateas, "Time in Video Games: A Survey and Analysis," *Simulation & Gaming - Simulat Gaming*, vol. 41, Jan. 2010, doi: 10.1177/1046878110375594.

[28] W. Zhao, J. Zhang, J. Yang, T. He, H. Ma, and Z. Li, "A novel joint biomedical event extraction framework via two-level modeling of documents," *Information Sciences*, vol. 550, pp. 27–40, Mar. 2021, doi: 10.1016/j.ins.2020.10.047.