Time-Oriented Question Answering from Clinical Narratives Using Semantic-Web Techniques

Cui Tao[1], Harold R. Solbrig[1], Deepak K. Sharma[1], Wei-Qi Wei[1], Guergana K. Savova[2], and Christopher G. Chute[1]

[1] Division of Biomedical Statistics and Informatics, Mayo Clinic, Rochester, MN
[2] Harvard Medical School, Boston, MA

Abstract. The ability to answer temporal-oriented questions based on clinical narratives is essential to clinical research. The temporal dimension in medical data analysis enables clinical researches on many areas, such as, disease progress, individualized treatment, and decision support. The Semantic Web provides a suitable environment to represent the temporal dimension of the clinical data and reason about them. In this paper, we introduce a Semantic-Web based framework, which provides an API for querying temporal information from clinical narratives. The framework is centered by an OWL ontology called CNTRO (Clinical Narrative Temporal Relation Ontology), and contains three major components: time normalizer, SWRL based reasoner, and OWL-DL based reasoner. We also discuss how we adopted these three components in the clinical domain, their limitations, as well as extensions that we found necessary or desirable to archive the purposes of querying time-oriented data from real-world clinical narratives.

1 Introduction

The rapid increase in the volume of electronic health records (EHR) available for research purposes provides new opportunities to create semantically interoperable healthcare applications and solutions for evidence-based medicine. An important aspect of EHR is the temporal ordering of clinical events. Time is essential in clinical research [20]. Exposing the temporal dimension in medical data analysis provides new research paths such as (1) uncovering temporal patterns at the disease and patient level to better understand the progression of a disease, (2) explaining past events such as the possible causes of a clinical situation, and (3) predicting future events such as possible complexities based on a patient's current status.

One important objective for enable meaningful use of EHR is to develop software applications "to realize the true potential of EHR to improve the safety, quality, and efficiency of care" [3]. In order to facilitate clinical researchers to expose the temporal dimension in medical data analysis, software platforms that allow users to ask free-form queries and retrieve temporal information automatically from clinical records are highly desired. First, the temporal information

P.F. Patel-Schneider et al.(Eds.): ISWC 2010, Part II, LNCS 6497, pp. 241–256, 2010.

interwoven in clinical narratives needs to be extracted and annotated to allow computer systems to be able to locate the information of interest. Second, temporal relations and assertions that are not explicitly expressed in the original documents need to be automatically inferred in order to enable the full capacity and true potential of secondary use of EHR for meaningful use. Third, temporal-oriented questions need to be captured in computer queries to query the annotated and inferred information.

The Semantic Web and the Web Ontology Language (OWL) [13] provide a suitable environment for modeling the temporal dimension of the clinical data, reasoning and inferring new knowledge, and querying for the information desired. The Semantic Web provides a standard mechanism with explicit and formal semantic knowledge representation, and automated reasoning capabilities. OWL is built on formalisms that adhere to Description Logic (DL) and therefore allows reasoning and inference. The Semantic Web Rule Language (SWRL) [23] can be used to add rules to OWL and enable Horn-like rules that can be used to infer new knowledge from an OWL based ontology and reason about OWL individuals. Once we have an ontology that can represent temporal assertions in the clinical domain precisely, we can annotate temporal expressions and relations with respect to the ontology and store the instances as RDF triples [17]. The information then become "machine-understandable". Tools and services such as reasoners, editors, querying systems, and storage mechanisms that have been developed by the Semantic Web community can be directly applied to the temporal data.

In this paper, we introduce a Semantic-Web based framework, which provides an API for querying temporal information from clinical narratives. The framework is centered by an OWL ontology called CNTRO (Clinical Narrative Temporal Relation Ontology), and contains three major components: time normalizer, SWRL based reasoner, and OWL-DL based reasoner. We also discuss how we adopted these three components in the clinical domain, their limitations, as well as extensions that we found necessary or desirable to archive the purposes of querying time-oriented data from real-world clinical narratives.

2 Related Work

Several approaches already exist for the modeling and query of temporal information. Most of these are research efforts that focus on temporal information stored in structured databases [32]. There are two existing temporal ontologies in OWL, the Time Ontology [29] and the SWRL Temporal ontology [25], the first of which is a general time ontology that defines basic time components and their relationships. And the second one is built for the SWRL Temporal Built-Ins library [24]. Both ontologies adopted Allen's Interval Based Temporal Logic [1], which provides a foundation of temporal logic for many temporal models. Tappolet and et al. [27] propose using time as an additional semantic dimension of data using RDF named graphs in combination with a temporal extension of the SPARQL query language called t-SPARQL. The SWRL Temporal Built-Ins library [24] defines a set of built-ins that can be used in SWRL rules to perform temporal operations and has been applied in clinical research such as the system described in [11]. These approaches, however, focus on the relationships between instances and intervals in time and it is not obvious how these relationships can be applied to actual events themselves.

There are also existing approaches that focus on the representations of free text narratives such as those encountered in clinical notes. Models such as Temporal Constraint Structure (TCS) [31] and the TimeML model [28] provide ways to represent temporal information in natural language. HL7 time specification [7] defines data types that can be used to specify the complex timing of events and actions such as those that occur in order management and scheduling system. While these models provide a good foundation, they are not currently compatible with OWL and other semantic-web based tools and do not support formal reasoning to infer new temporal knowledge.

3 Clinical Narrative Temporal Relation Ontology

We have developed an ontology in the Web Ontology Language (OWL) format for modeling temporal information in clinical narratives, and evaluated this ontology using real-world clinical notes [26]. In this section, we briefly introduce our OWL ontology for temporal relation reasoning in clinical narratives, which we call the Clinical Narrative Temporal Relation Ontology (CNTRO)[1]. This ontology can model the temporal information found both in structured databases and in natural-language based clinical reports. We investigated the existing conceptual models for temporal information cited in the previous section. CNTRO was developed based on these previous experiences combined with new ontological specifications that fit the needs of natural-language based clinical reports. We decided to first build a stand-alone model based on our requirements, which is what is described in this paper. Subsequent work will involve the integration of CNTRO and existing ontologies that cover time-related components such as the Time Ontology in OWL [29], and Basic Formal Ontology (BFO) [2].

Figure 1 shows the graphical view of the ontology. OWL classes are represented by a rectangles with rounded corners and data types are represented by an ovals. Subclass relationships are represented by hollow-headed arrows and object and data properties by solid-headed arrows.

The class, *Event* represents an occurrence, state, perception, procedure, symptom or situation that occurs on a time line in clinical narratives.

The *Time* class is the superclass of all the OWL temporal representation classes: *TimeInstant, TimeInterval, TimePhase*, and *TimePeriod*. An OWL *TimeInstant* is a specific point of time on the time line. In clinical reports, a time instant can be represented in different granularities such as year, month, and day. A time instant may also be represented in different formats. We implemented a normalizer that converts commonly used time notations to the XML dateTime

[1] http://www.cntro.org

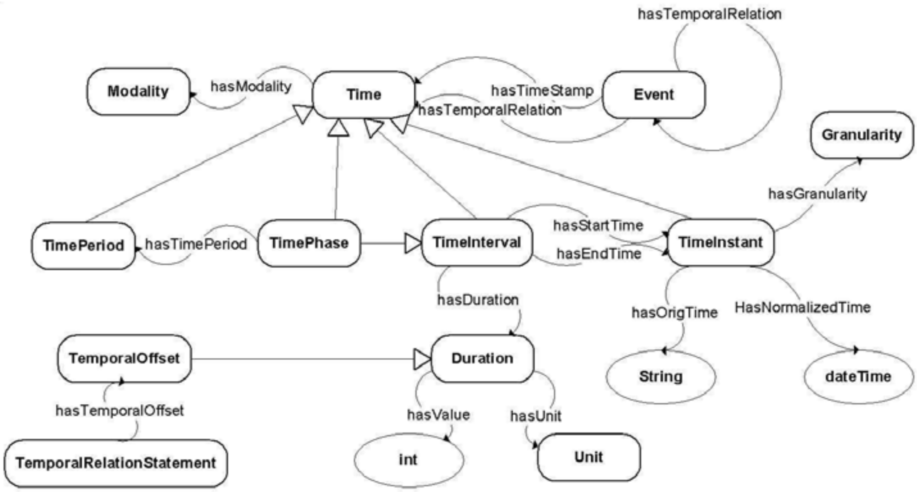

Fig. 1. A Graphical View of Clinical Narrative Temporal Relation Ontology

format. In the ontology, we defined two data properties *hasOrigTime* and *hasNormalizedTime* that keep track of the time instant in its original form and in the normalized form respectively. An OWL *TimeInterval* represents a duration of time. It could have two relations (OWL object properties), *hasStartTime* and *hasEndTime*. Each of them links to instances of *TimeInstant*. A *TimeInterval* could also have a *Duration*. An instance of the *Duration* class represents the time length of a *TimeInterval*. We use an OWL data type property *hasValue* and an OWL object property *hasUnit* to describe a *Duration*. Many clinical events recur periodically. Adopted and modified from the HL7 time specification [7], two OWL classes, *TimePhase* and *TimePeriod*, are defined in CNTRO to represent intervals of time that recur periodically. A *TimePhase* represents each occurrence of the repeating interval and a *TimePeriod* specifies a reciprocal measure of the frequency at which the *TimePhase* repeats. The class *TimePhase* is a subclass of *TimeInterval*, therefore, we can also specify a *StartTime*, an *EndTime*, and a *Duration*. In addition, a relation (OWL ObjectProperty), *hasTimePeriod*, is defined to specify the relation between a *TimePhase* and a *TimePeriod*. For example, "every 8 hours for 10 days starting from today" is a *TimePhase*. Its *StartTime* is "today". Its *Duration* is "10 days". And its *TimePeriod* is "every 8 hours". We also define the certainty of a *Time* instance. For example, a physician can describe a time notation with ambiguities such as "early next week" and "in approximately two weeks". In the CNTRO ontology, we defined a class called "Modality" which serves as a flag to indicate whether a time representation is approximated or not.

We can define the temporal relations between two events, or between an event and a time instance using the object property *hasTemporalRelation* and its subproperties. We also use Allen's temporal logic operators when defining our temporal relation properties: equal, before, after, during, meet, start, finish, and during. We have also defined their logical characteristics. For example, *before* is a transitive property, and its inverse property is *after*.

We can also use *TemporalRelationStatement* class to describe temporal relations between two events or between an event and a *Time* instance. The *TemporalRelationStatement* class is a sub-class of *rdf:Statement*, we can define temporal subject, object, and predicate of a *TemporalRelationStatement*. Using *TemporalRelationStatement* to describe a temporal relation enables defining properties of the relation by reification. For example, we can add an offset time frame to the relation by using an OWL object property called *hasTemporalOffset*. The domain of *hasTemporalOffset* is *TemporalRelationStatement* and the range of it is *Duration*. This offset defines the relative timing of a pair of events. In order to model the sentence "patient's bilirubin is elevated 2 weeks after the second cycle of chemotherapy", for example, we can use a *TemporalRelationStatement* to represent "patient's bilirubin is elevated" (object) *after* (predicate) "the second cycle of chemotherapy" (subject), and then add "2 week" as an instance of *TemporalOffset* to this *TemporalRelationStatement* instance.

We compared the expressiveness capabilities of the CNTRO ontology with the two existing temporal ontologies in OWL: the Time ontology [29] and the SWRL Temporal ontology [25]. Since these two ontologies are designed only for structured data in databases, they mainly focus on timing events with points anchored in absolute time. To cover the temporal assertions in natural-language based clinical narratives, we have added the following major expressiveness capabilities to the CNTRO ontology. (1) ***Periodic Time Interval.*** In clinical narratives, there are many events that recur periodically. It is important to be able to represent periodic time intervals. Two OWL classes, TimePeriod and TimePhase, have been defined to represent periodic time intervals in the CNTRO ontology. (2) ***Relation between Two Events.*** In many cases in clinical notes, physicians describe the relations between two events without indicating the time stamps of the events, i.e., (*patient's bilirubin is elevated after the second cycle of chemotherapy*). While the other two OWL ontologies defines that the temporal relations are only between Time entities themselves, CNTRO is able to capture this kinds of qualitative temporal relationships. (3) ***Time Offset.*** The CNTRO ontology defines a *TemporalOffset* class which enables representing the time offset of a relation using reification. (4)***Relative Time.*** Relative time such as "today", "tomorrow", "two months ago", or "in 3 weeks" is very commonly used in clinical reports. The CNTRO ontology captures the relative time information in its original form and at the same is able to represent the calculated absolute time in the normalized form. (5) ***Uncertainty.*** Often temporal information is represent with uncertainty in clinical notes. CNTRO offers a property–*hasModality* to track of the uncertainty to make sure it can be taken into consideration in answering temporal questions.

4 Temporal Information Reasoning

CNTRO provides a conceptual model to represent temporal relations in clinical narratives. A lot of time qualitative and quantitative temporal relationships, however, are expressed implicit in the event occurrences. The answers of many time-oriented questions are not necessarily stated explicitly in clinical narratives, but rather need to be inferred. For example, here are three sentences from one patient's clinical notes: (1)*Patient's INR value is below normal* (Event_1) today. (note date: 01/26/07) (2)He has had the *chills and body aches* (Event_2) before *the abnormal test.* (Event_3)" (note date: 01/26/07) (3) On Jan. 30, 2007, patient started *Coumadin dosing plan of 1.0 mg* ((Event_4)).(note date: 02/09/07) To answer the question "did the patient experience body aches before the he started the Coumadin dosing plan?", we actually need a few different steps of inferences. We know that Event_1 has a time stamp "today" (time instant); Event_2 is *before* Event_3; and Event_4 has a time stamp which is a time interval that has start date "Jan. 30, 2007". We first infer that the date of "today" for Event_1 is the note date, which is "01/26/07". We then infer that Event_3 actually refers to Event_1. Therefore we know that Event_2 is before Event_1, which happened on "01/26/07". Hence, we know that Event_2 is before "01/26/07". Now we need to compare "01/26/07" and "Jan. 30, 2007" which is the Event_4's time stamp. In order to do that, we need to normalize the two dates, and infer that "01/26/07" is before "Jan. 30, 2007". Since that Event_2 is before "01/26/07", which is before "Jan. 30, 2007", which is the start time of Event_4, we then can finally infer that Event_2 is before Event_4.

This simple example illustrates how reasoners can help to infer temporal relations. In this section, we discuss three major components we need to do temporal relation inferences.

4.1 Temporal Representation Normalization

Temporal information in clinical text can be expressed in different ways [32]. In order to infer temporal relations in clinical narratives, our first step is to normalize the time expressions. Because the clinical records we are working with are from the US based Mayo Clinic, this research focuses on conversion of commonly used US temporal notations [4] to the xsd DateTime Data Type format [30]. We used the information extraction technology developed by the Brigham Young University (BYU) Data Extraction Group (DEG) [6] to recognize different time notations. The DEG group has developed a set of libraries that recognize when the same concept when represented in different formats, and we make use of their time recognition component to identify different representations of the same time. We then normalize the format and convert it into the xsd DateTime format associating both the original and normalized time with an instance of the *TimeInstance* class. The normalizer can also recognize the granularity of a time expression. The defined six different units of measures to represent different levels of granularity: year, month, day, hour, minute, and second. In this particular paper, the finest granularity we cover is day.

Temporal references often occur as relative terms within clinical text. Terms such as "today", "tomorrow", "last month", "two years ago", and "in two months" permeate the clinical document. The normalized form of a relative temporal reference can often be inferred from to its relationship to other absolute and relative temporal references. As an example, "today" is a relative expression since its value depends on the document context. As we always know date when a clinical narrative was written we can use it to convert "today" into an absolute equivalent date. Other relative temporal references can be converted to absolute equivalents with an accompanying granularity. As an example, if a clinical document was recorded on 2010-06-08, we can infer that "in 2 days" corresponds to 2010-08-10, with a granularity of *day*. The SWRL temporal built in library [24] provides functions to calculate to a time reference by adding or subtracting a duration from a given time point. Section 4.3 discusses how we adopt it in detail.

4.2 OWL DL Reasoning

Logical Characteristics of Properties. We leverage the logical definition properties to infer more temporal relations between events. For example, *before* is defined as being transitive, meaning that, if that event A is stated as occuring *before* event B, and event B *before* event C, we can infer that event A occurrs *before* event C. *Before* and *after* are defined as inverse properties. Therefore, given that event A is *before* event B, we can infer that event B is *after* event A, and vice versa. *equal* is defined as a symmetric property, meaning that, when event A is described as being *equal* with event B, we can infer that B is also *equal* with A. The temporal relations can be semantically defined using SWRL rules or computed using SWRL Built-Ins, which we will discuss in the next section.

CNTRO also provides the capability to define time offsets for temporal relations. Based on these time offsets, more temporal relations could be inferred. The RDF quads below provide an example

```
S1 e1 before e2
S1 hasTemporalOffset d1 (3 days)       [e1 occurred 3 days before e2]
S2 e2 before e3
S2 hasTemporalOffset d2 (2 days)       [e2 occurred 2 days before e3]
S3 e2 after e4
S3 hasTemporalOffset d3 (2 days)       [e2 occurred 2 days after e4]
```

Since *before* and *after* are transitive properties, we can use a reasoner such as Pellet [14] to infer that event *e1* is also *before* event *e3*. But Pellet does not provide the reasoning power to infer the temporal relation between events *e1* and *e4*. Based on the temporal offsets, however, we can calculate the time interval between these events using a pair of inverse operators α and β to calculate time interval based on temporal offsets, where α is used when the temporal relation is *after* and β is used when the temporal relation is *before*. To calculate the time interval between events *e1* and *e4*, we then have an operation, β(3 days)α(2 days). Since α and β are inverse operators, the result of this operation is β(1 day) meaning *e1* occurred 1 day before *e4*.

Restriction Assertions. With the temporal relations defined in CNTRO, we can use OWL restrictions to define known temporal relationships between different kinds of events. For example, we want define that a treatment of a condition must happen after it has been diagnosed. We define the temporal relations between the two SNOMED CT concepts: *CancerChemotherapy* and *CancerDiagnosisBasedOnClinicalEvidence* as

```
Class(sct:CancerChemotherapy partial
  restriction(CNTRO:after
    someValuesFrom (sct:CancerDiagnosisBasedOnClinicalEvidence)))
```

This definition allows as to restrict that a cancer chemotherapy must happen after a cancer diagnosis based on clinical evidence.

The above definition, however, is slightly different than what we need. A patient could have more than one diagnoses and treatments. We want to be able to specify that a treatment of a condition must happen after **the** diagnosis for this particular condition. We must consider the two relations *Treatment treats ConditionWithDiagnosis* and *Treatment after ConditionWithDiagnosis* together to ensure the correct semantic meaning. We need to be able to add the temporal relation property as a qualifier of the another relation. So that we can link a restriction to the class description to define a class of individuals x for which holds that if the pair (x,y) is an instance of P (the property concerned), then y should have certain temporal relation with x. So this is our preferred way to represent our example:

```
Class(sct:CancerChemotherapy partial
  restriction(treats
    CNTRO:after (sct:CancerDiagnosisBasedOnClinicalEvidence)))
```

This restriction describes the temporal qualification of a relation, if an instance of *CancerChemotherapy* a is for an instance of *CancerDiagnosisBasedOnClinicalEvidence* b, then a must happen after b. This definition can be described using SWRL rules, which we will discuss in the next section.

Semantic Definition of Concepts. With OWL DL, we can formally define clinical events or clinical-related temporal periods with temporal assertions, such as “infection after injection” and “before procedure”. For example, SNOMED CT defines that “infection after injection” is a “infection as complication of medical care” that is after “injection”. Using OWL DL, we define the *InfectionAfterInjection* class as fellow:

```
Class(InfectionAfterInjection partial
  intersectionOf
    (restriction(CNTRO:after someValuesFrom (Injection))
    InfectionAsComplicationOfMedicalCare))
```

With formal semantic definitions of clinical events, we can use the reasoners to automatically identify certain time-related events from patient records. This capability will potentially bring benefits to high throughput phenotyping, GWA (genome-wide association) studies, clinical trials, and epidemiology studies.

4.3 SWRL-Based Reasoning

SWRL Temporal Built-In Library. The SWRL Temporal Built-Ins Library is one of the SWRLTabBuiltInLibraries [24]. It defines a set of builtins that can be used in SWRL rules to perform temporal operations. It works with temporal information in the normalized form. Given two normalized time stamps, the Built-Ins provide basic functions such as calculating the durations, and comparing the two time stamps and checking if they satisfy certain temporal relations. It also can compare two durations and check if one is less than, equal to, or greater than the other. In addition, the Temporal Builtins provides an *add* function, which can calculate a new time stamp by adding (or subtracting) a duration from a given time stamp.

The temporal Builtins provides us the basic function blocks to build our temporal reasoner. After the temporal data has been normalized, many more information can be inferred or calculated using the function blocks. For example, with the *add* function, we can calculate the start/end time of a time interval given the end/start time and its duration. We can also calculate the time stamp of an event, given the time stamp of another event, and the temporal relation with time offset of the two events.

SWRL RuleML. SWRL is designed based on the combination of OWL DL and the unary/binary Datalog RuleML sub-language [23]. We can use SWRL to add semantic assertions and enable Horn-like rules that can be used to infer new knowledge from an OWL ontology and reason about OWL individuals. A rule composed by two or more shared variables is easily expressed in Datalog and corresponding decidable subsets of rule based languages. However, such role chains is hard to be expressed in OWL DL [8]. SWRL generalized OWL by allowing arbitrary patterns of variables and property conditionals expressions.

Using SWRL and the temporal relations defined in CNTRO, we can further define time events with complex temporal assertions and/or with more than two shared variables. For example, we can define that for a valid time interval, its start time must before its end time by the following rule:

```
TimeInterval(?t)^hasStartTime(?t, ?s)^hasNormalizedTime(?s, ?ns)
^hasEndTime(?t,?e)^hasNormalizedTime(?e, ?ne)^before(?ns,?ne)
--> ValidTimeInterval(?t)
```

We can also define temporal relation properties such as *meet*, *during*, *overlap*, *finish*, and *start*. For example, the temporal relation property *during* is defined as follow:

```
Event(?a1)^hasTimeStamp(?a1,?t1)^TimeInterval(?t1)^
hasStartTime(?t1,?s1)^hasNormalizedTime(?s1,?ns1)
hasEndTime(?t1,?e1)^hasNormalizedTime(?e1,?ne1)
Event(?a2)^hasTimeStamp(?a2,?t2)^TimeInterval(?t2)^
hasStartTime(?t2,?s2)^hasNormalizedTime(?s2,?ns2)
hasEndTime(?t2,?e2)^hasNormalizedTime(?e2,?ne2)
^before(?ns1,?ns2)^after(?ne1,ne2)
--> during(?a2,?a1)
```

```
Event(?a1)^hasTimeStamp(?a1,?t1)^TimeInterval(?t1)^
hasStartTime(?t1,?s1)^hasNormalizedTime(?s1,?ns1)
hasEndTime(?t1,?e1)^hasNormalizedTime(?e1,?ne1)
Event(?a2)^hasTimeStamp(?a2,?t2)^TimeInstant(?t2)^
hasNormalizedTime(?t2,?nt2)^
^before(?ns1,?nt2)^after(?ne1,nt2)
--> during(?a2,?a1)
```

We assume that if an event *A* includes another event *B*, event *A* must be associated with a time interval. Event *B*, however, could be associated with either a time instant or a time interval, each defined by one of the above rules. These temporal operators can also be expressed using SWRL Built-Ins that connect to Java methods.

We can combine the SWRL Built-Ins predicates and operators with SWRL rules to define clinical events and concepts. For example SNOMED CT has a concept "premature labor after 22 weeks but before 37 completed weeks of gestation without delivery", we can use SWRL rule expression to define the temporal part as:

```
Event(?p)^hasTimeStamp(?p, ?t) ^ hasDuration(?t, ?d)
^temporal:durationLessThan('154', ?d, temporal:Days)
^temporal:durationGreaterThan('259', ?d, temporal:Days)
```

In the above expression uses the two operators in *durationLessThan* and *durationGreaterThan* from SWRL temporal builtins to check if the duration of the event falls in the range specified in the concept. Since both the SWRL temporal builtins and CNTRO do not support the level of granularity on *week*, we have to convert *22 weeks* and *37 weeks* to 154 *days* and 259 *days*.

5 Implementation Status

We have designed and built a framework that embeds normalization, DL-based reasoning, and SWRL-based reasoning. The framework adopted the temporal computation components from the SWRL Temporal Built Ins library and uses Pellet [14] as the reasoning engine. It provides a query API for users to query data represented with respect to CNTRO. General search API parameters are:

- **findEvent(searchText)** returns a list of events that match the searching criteria. Currently we look for events based on text search. We are working on connecting our reasoning framework with Mayo Clinic's Text Analysis and Knowledge Extraction System (cTAKES) [12]. cTAKES can annotate clinical events with respect to standard ontologies such as SNOMED CT [21] (for clinical terms) or RxNorm [18] (for drug names). It annotates named entities expressed in different ways but have the same semantic meanings using the same concept code. We can then search by concept codes or labels instead.
- **GetEventFeature(event, featureflag)** returns a specific time feature for a given event. The parameter featureflag indicates which time feature the

user wants to retrieve: start time, end time, note taken time, or event time. All the time will be returned in the normalized format. If the specific time was not stated in the original file explicitly, it will call the reasoner and check if the time can be inferred. **Sample query**: When was the patient diagnosed with diabetes? When was the patient started his chemotherapy?
- **getDurationBetweenEvents(event1, event2)** returns the time interval between two events. The duration of the interval is either retrieved directly, calculated, or inferred from temporal relationships with offsets. **Sample query**: How long after the patient was diagnosed colon cancer did he start the chemotherapy?
- **getDuration(event)** returns the duration of a given event. The duration can be either retrieved directly or calculated. **Sample query**: How long did the symptoms of rectal bleeding last?
- **getTemporalRelationType(event1, event2)** returns the temporal relations between two events if it can be retrieved differently or inferred. **Sample query**: Was the PT scan after the colonoscopy?
- **getTemporalRelationType(event1, time)** returns the temporal relations between an event and a specific time if it can be inferred or retrieved. **Sample query**: Is there any behavior change within a week of the test?
- **getEventsTimeline(events)** returns the order (timeline) of a set of events. Optionally, when the order of the given list of events cannot be completely resolved, it returns a set lists with those events that cannot be sorted within the group. **Sample query**: What is the tumor status timeline as indicated in the patient's radiology note? What is the treatment timeline as recorded in oncology notes? When was the first colonoscopy done When was the most recent glucose test?

This temporal reasoning framework is an ongoing process. We are working on implementing and improving the features of the API, and evaluating the API with real world clinical data.

6 Discussions

Instant vs. Interval. Whether to view time as instants or intervals is a debate among a lot of researchers [32]. On one hand, a time instant can be viewed as a time interval with a very short duration. On the other hand, a time interval is a time instant on a coarse level of granularity. In medical text, both time instants and time intervals are used to describe clinical events. For example, a clinician may state that "patient's last cycle of chemotherapy was on Jan. 19", or "patient's last cycle of chemotherapy started from Jan. 10 and ended on Jan. 19". Currently we annotate the time stamp of an event simply based on the expressions themselves. When there is only one time expression stated, we consider it as a time instant. If duration, and/or start and end time were stated, we consider it as a time interval. Therefore, we consider that "Jan. 19" is a time instant with granularity *Day* whereas "started from Jan. 10 and ended on Jan. 19" is a time interval with both start time and end time indicated.

One might argue, however, that a cycle of chemotherapy should be a process with a duration instead of an occurrence that just happens on a specific point of time. We are currently investigating on how to further classify and specify events into different categories with different temporal characteristics. We then will be able to annotate the temporal information of an event based on the temporal characteristics of the event itself, instead of based on the temporal expressions used in the original documents. For example, for processes like chemotherapy or surgery, we use time intervals. But for occurrences like checking-in, we use time instants. Basic Formal Ontology and Medical Ontology [2] has defined different kinds of occurrences and process entities. We plan to adopt and expand the classes defined by BFO to our CNTRO ontology, so that the temporal information can be more properly annotated.

Temporal Uncertainty and Temporal Imprecision. There are different kinds of uncertainties we have encountered during both the annotation process and the reasoning process.

One kind of uncertainty is from the original source. CNTRO has defined a property called *hasModality* to capture uncertainties specified explicitly in the original documents. For example, "in approximately two weeks" or "about 3 hours", each is an approximated temporal expression with uncertainties. Temporal relations that are inferred based on this kind of temporal expressions will also be returned to users as approximated.

In clinical text, each time expression is stated on a certain level of granularity. But is that level of granularity sufficient enough for inferring temporal relations or calculate a duration? One example would be to get the duration between an event happened on Jan. and an event happened on June. Is that 5 months, 6 months, or 7 months? Another example is that an event A has time stamp "Jan", and an event B has time stamp "Jan 16". The reasoner could not infer a certain temporal relation between these two events. This kind of uncertainties was major caused by temporal imprecisions.

We also found that temporal information in clinical text can be expressed in a coarse notion that it is hard to use one of the pre-defined levels of granularity to describe it, i.e., "early next year", "middle of next week", "short after 11:30 PM", or "immediately after admission". This kind of imprecisions brings us problems for uncertainties on temporal relations and durations too. For example, given that an event A has a time stamp "short after 11:30PM on Jan 16", and an event B has a time stamp "Jan 17", how confident can we say that event A is before event B?

In addition, sometimes one temporal expression can have different interpretations. For example, for the sentence "patient's last cycle of chemotherapy was on Jan. 19", there might be three different interpretations: (1) patient's last cycle of chemotherapy STARTED on Jan. 19; (2) patient's last cycle of chemotherapy ENDED on Jan. 19; or (3) patient's last cycle of chemotherapy STARTED and ENDED on Jan. 19. If we can specify the common duration of a cycle of chemotherapy, it might be helpful to disambiguate the confusions. For example, if we know the event usually lasts a few hours, but not a few days, we could interpret that patient's last cycle of chemotherapy STARTED and ENDED on Jan. 19.

How to describe the uncertainty in a systematic way while still support meaningful reasoning powers is a non-trivial problem. While OWL can provide means for including numeric uncertainty measures or level of uncertainties as data type properties, there is no standardized way of representing uncertainties. In order to adequately represent uncertainties in OWL, some language extension is necessary. For example, previous research has focused on extending OWL DL with fuzzy set theory [10,22], or using Bayesian networks as the underlying reasoning mechanism and probabilistic model [5,16]. We are currently investigating on adopting this previous work and using OWL to represent temporal uncertainties. In addition, we believe it will be useful to use ranges to represent imprecise temporal notions and currently working on extend the CNTRO ontology to reflect it.

Negation. SWRL and OWL's monotonicity assumption determines that negation as failure is not supported. But in practice, we need to have a *not* operator in both annotation and reasoning. In many cases, clinicians use negations of temporal relations in clinical narratives, such as "no later than", "not during", and "not before". Without a not operator, new temporal relation properties such as *not_before*, *not_after* have to been introduced and semantically defined, like what the SWRL Temporal Built-In Ontology does.

Limitations with SWRL Built-Ins. While SWRL Built-Ins provide a powerful extension mechanism that allows user-defined methods to be used in rules, and serve as important function blocks in our temporal relation reasoning framework, we found there are some limitations when using them. First, the Built-Ins do not use an input-output designation mechanism. Built-ins can assign (or bind) values to arguments. The implementation of the rule engine must detect the unbound arguments and assign values to them. The types or the positions of the unbound arguments cannot be defined through SWRL rules, therefore errors cannot be detected easily before run time. Therefore, we provided our own API for queries.

In addition, the SWRL Temporal Built-Ins implementation is not available as a stand-along program library yet. We have investigated two ways to leverage the Built-Ins library: (1) using the Protégé SWRL tab [15], and (2) using Pellet reasoner for SWRL Built-Ins. The first one can only be used in Protégé environment and the second has limited access to temporal operations. In our framework, we leveraged basic temporal Java classes implementation that comes with SWRL tab plug-in for Protégé, such as Instant, Period and Temporal to compute basic features and relations among events in patient's clinical note.

Timing-Event-Dependent Change It is important to monitor the changes between two time points or two timing events. For example, in "Most recent ultrasound in May 2007 showed no change comparing to Nov last year", we can annotate two timing events, "ultrasound in May 2007" and "ultrasound in

Nov last year". But with the current model, it is hard to annotate "no change" between these two events. BFO has explored two ways to representing changes: by comparing the discrepancies among the qualities at different time instants, or by capturing the continuous dynamic change over an interval of time. While measurement of change has been a topic widely covered by many researchers, currently there is no standard way for modelling it in OWL. OWL's monotonicity assumption precludes modelling the changes of property values over time without significant extra effort to circumvent the imposed constraints [9].

7 Conclusions and Future Work

In this paper, we introduce a Semantic-Web based framework for querying and inferring temporal information from clinical narratives. We have built an OWL ontology that models temporal information such as timing events, time instants, time intervals, durations, and temporal relations. Based on this ontology, temporal information in clinical narratives can be annotated and represented in RDF. This ontology also provides foundation pillars for us and users to define concepts and relations in the temporal aspects. Our framework embedded OWL DL-based reasoning, SWRL-based reasoning, and the SWRL Temporal Built-Ins library, combined these tools seamlessly to fit the needs of time-oriented question answering and inference from clinical narratives.

Several directions remain to be pursued. First, we would like to connect the reasoning framework to Mayo Clinic's Text Analysis and Knowledge Extraction System (cTAKES) [12]. We will extend and improve cTAKES and use it as an automatic annotator for temporal information [19] and annotate information with respect to the CNTRO ontology. We want to scale up the data collection and investigate more on reasoning temporal information in clinical narratives. We would also like to address the consistency issues and object identification problem over heterogeneous sources. Second, we would like to extend the CNTRO ontology and embed more time-related semantic assertions as discussed in Section 4.2. We will also embed the SWRL rules discussed in Section 4.3 into the ontology itself. In addition, we will explore how to leverage the capabilities of Rule Interchange Format (RIF) and OWL2 for temporal information definition and reasoning. Third, we want to extend the CNTRO so that we can capture data with uncertainty and imprecision better as discussed in Section 6. Last, but not least, we want to implement a user-friendly user interface for healthcare providers and clinical researchers to query the time-related information in clinical narratives.

Acknowledgement. This research is supported by the National Science Foundation under Grant #0937060 to the Computing Research Association for the CIFellows Project, the ONC Strategic Health IT Advanced Research (SHARP) award under Grant #90TR0002-01, and the Mayo Clinic eMERGE study under Grant #U01-HG04599.

References

1. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM 26(11), 832–843 (1983)
2. Basic Formal Ontology (BFO), http://www.ifomis.org/bfo
3. Blumenthal, D., Tavenner, M.: The "meaningful use" regulation for electronic health records. The New England Journal of Medicine (NEJM) 363(6), 501–504 (2010)
4. Date and time notation by us, http://en.wikipedia.org/wiki/Date_and_time_notation_by_country#United_States
5. Ding, Z., Peng, Y.: A probabilistic extension to ontology language OWL. In: Proceedings of the 37th Hawaii International Conference on System Sciences, HICSS-37 (2004)
6. Embley, D.W., Campbell, D.M., Liddle, S.W., Smith, R.D.: Ontology-based extraction and structuring of information from data-rich unstructured documents. In: Proceedings of the 7th International Conference on Information and Knowledge Management (CIKM 1998), Washington D.C, pp. 52–59 (November 1998)
7. HL7 time specification, http://www.hl7.org/v3ballot/html/infrastructure-datatypes/datatypes.htm
8. Horrocks, I., Patel-Schneider, P.F.: A proposal for an owl rules language. In: Proceedings of the Thirteenth International World Wide Web Conference (WWW 2004), Manhattan, New York (2004)
9. Matheus, C.J., Baclawski, K., Kokar, M.M., Letkowski, J.J.: Using swrl and owl to capture domain knowledge for a situation awareness application applied to a supply logistics scenario. In: Adi, A., Stoutenburg, S., Tabet, S. (eds.) RuleML 2005. LNCS, vol. 3791, pp. 130–144. Springer, Heidelberg (2005)
10. Mazzieri, M., Dragoni, A.F.: A fuzzy semantics for the resource description framework. In: da Costa, P.C.G., d'Amato, C., Fanizzi, N., Laskey, K.B., Laskey, K.J., Lukasiewicz, T., Nickles, M., Pool, M. (eds.) URSW 2005 - 2007. LNCS (LNAI), vol. 5327, pp. 244–261. Springer, Heidelberg (2008)
11. O'Connor, M.J., Shankar, R.D., Parrish, D.B., Das, A.K.: Data integration for temporal reasoning in a clinical trial system. International Journal of Medical Informatics 78(1), S77–S85 (2009)
12. cTAKES on open health natural language processing (OHNLP) consortium, http://www.ohnlp.org
13. OWL Web Ontology Language Reference, http://www.w3.org/TR/owl-ref/
14. Pellet: Owl 2 reasoner for java, http://clarkparsia.com/pellet/
15. The Protégé Ontology Editor, http://protege.stanford.edu/
16. PR-OWL: A bayesian extension to the OWL ontology language, http://www.pr-owl.org/
17. Resource description framework (rdf), http://www.w3.org/RDF/
18. RxNorm, http://www.nlm.nih.gov/research/umls/rxnorm/
19. Savova, G., Bethard, S., Styler, W., Martin, J.H., Palmer, M., Masanz, J., Ward, W.: Towards temporal relation discovery from the clinical narrative. In: Proceedings in the American Medical Informatics Association (AMIA) Annual Symposium, San Francisco, California (November 2009)

20. Shahar, Y.: Timing is everything: Temporal reasoning and temporal data maintenance in medicine. In: Proceedings of Artificial Intelligence in Medicine. Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making (AIMDM 1999), Aalborg Denmark, pp. 30–46 (June 1999)
21. Systematized nomenclature of medicine–clinical terms (SNOMED CT), http://www.snomed.org
22. Stoilos, G., Stamou, G.: Extending fuzzy description logics for the semantic web. In: Proceedings of the 3rd International Workshop on Owl: Experiences and Directions (2007)
23. A Semantic Web Rule Language Combining OWL and RuleML, http://www.w3.org/Submission/SWRL/
24. SWRL temporal built-in library, http://protege.cim3.net/cgi-bin/wiki.pl?SWRLTemporalBuiltIns
25. The SWRLTab's valid-time temporal ontology, http://swrl.stanford.edu/ontologies/built-ins/3.3/temporal.owl
26. Tao, C., Wei, W.-Q., Savova, G., Chute, C.G.: A semantic web ontology for temporal relation inferencing in clinical narratives. In: Proceedings of the American Medical Informatics Association (AMIA) 2010 Annual Symposium, Washington DC (November 2010) (accepted)
27. Tappolet, J., Bernstein, A.: Applied temporal rdf: Efficient temporal querying of rdf data with sparql. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) ESWC 2009. LNCS, vol. 5554, pp. 308–322. Springer, Heidelberg (2009)
28. Markup language for temporal and event expressions, http://www.timeml.org/site/index.html
29. Time ontology in OWL, http://www.w3.org/TR/owl-time/
30. XML Schema Date/Time Datatypes, http://www.w3.org/TR/xmlschema-2/
31. Zhou, L., Melton, G., Parsons, S., Hripcsak, G.A.: A temporal constraint structure for extracting temporal information from clinical narrative. Biomedical Informatics 39(4), 424–439 (2006)
32. Zhou, L., Parsons, S., Hripcsak, G.: The evaluation of a temporal reasoning system in processing clinical discharge summaries. JAMIA 15(1), 99–106 (2008)