Published in final edited form as:
J Biomed Inform. 2013 October ; 46(5): 882–893. doi:10.1016/j.jbi.2013.07.003.

Semantator: Semantic Annotator for Converting Biomedical Text to Linked Data

Cui Tao[a,1], **Dezhao Song**[a,b], **Deepak Sharma**[a], and **Christopher G. Chute**[a]

[a]Division of Biomedical Statistics and Informatics, Mayo Clinic, Rochester, MN 55905

[b]Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA 18015

1. Introduction

As recent surveys indicated, more than 80% of patients seek health information on the Internet [3]; more than 70% of physicians regularly search online for medical or professional updates [19]. Approximately 80% of health care data, as well as the ever-growing data online, however, consist of unstructured narratives [14, 18]. Effciently querying and browsing data embedded in these biomedical documents is an important and challenging task. The unstructured nature of these text-based documents brings to light an inherent problem: locked within these documents lies an extraordinary amount of key biomedical knowledge and clinical data, which can hardly be leveraged without intensive manual work. Traditional search engines such as Google can return users the potential documents of interest based on keywords. Users still have to, however, read through the returned documents until the information of interest is located. In addition, search engines usually return hundreds of thousands of links, many of which are not relevant to users' search.

One approach to facilitate browsing and querying biomedical text is to convert the plain text into an annotated web of data, i.e., to convert data originally in free text into structured formats with defined meta-level semantics. Manual annotation may not be realistic due to the large volume of text that needs to be processed. Fully automatic approaches for semantic annotation do not always give satisfying results. Semi-automatic data annotation is, therefore, an attractive alternative. Semi-automatic annotation supports information from biomedical text to be automatically extracted and annotated with manual on refining the annotations.

To support semi-automatic annotation, we developed Semantator. Semantator is a user-friendly, semantic-web-oriented environment for annotating data of interest in biomedical documents with respect to domain ontologies. Domain ontologies have been used in information technology to provide semantic definitions of a particular domain, which enable automated agents to perform queries intelligently and infer new knowledge. An ontology

[1]Contact author: tao.cui@mayo.edu.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

includes a set of classes and their relationships (e.g., class hierarchies and predicates). Semantator provides an environment to link data embedded in text to ontology concepts by using semantic annotation. Information of interest from a document can be annotated as an instance of an ontology class to obtain all the semantic definition of that class. In addition, relations between instances can be created using the predicates (properties) defined in the ontology. The annotation results are saved in the Resource Description Framework (RDF) [21] format, which provides a standard way for data sharing and exchange and enables querying and browsing the data using the SPARQL query language [24]. In addition, Semantator also provides an interface where users can compare annotations done by different curators or annotation tools, leverage semantic web technologies for inferences, and detect conflicts in annotations.

More specifically, Semantator is implemented as a Protégé [2] plug-in, which allows users to view the original documents, the ontology used for an notation, and the annotation results in the same environment. Semantator provides two modes: 1) manual annotation, and 2) semi-automatic annotation. In the manual annotation mode, an expert can choose an annotation schema (a domain ontology), open a document to be annotated, highlight different pieces of information to be annotated, and then mark which ontology concepts the information belongs to. For each highlighted piece of data, the system will generate class instances according to the annotation and display different class instances in different colors. Relationships between instances can also be created using the properties defined in the domain ontology. For the semi-automatic annotation mode, Semantator provides an Application Programming Interface (API), which provides the option to connect the Semantator annotation environment to state-of-the-art or customized information extraction or semantic annotation tools. Human curators can review the automatic annotation results in the Semantator environment and modify them as needed.

The Semantator has been released through our web site: http://informatics.mayo.edu/CNTRO/index.php/Download_Semantator. In our previous publication [23], we reported the basic functionalities of Semantator: preliminary implementation of the manual annotation mode; and semi-automatic annotation using the clinical Text Analysis and Knowledge Extraction Systems (cTAKES) [22] and the NCBO annotator [16] (Section 3). This manuscript extends our previous work by introducing two new major functionalities: 1) rule-based extraction capacity (Section 4) and 2) the annotation result comparison function (Section 5). We analyze and illustrate the benefits of using semantic web technologies on the Semantator annotated data (Section 6). We have also conducted a functionality evaluation (Section 7.1) and applied Semantator in a real clinical research application as a case evaluation (Section 7.2). The evaluation results indicate that Semantator can successfully conduct the annotation tasks as designed. We have received much positive feedback and suggestions from the community, based on what we have already improved and will continually improve the functionalities of the tool (Section 8).

2. Related Work

2.1. Annotation Systems

Andrews et al [4] has reviewed a number of annotation systems and classified them into four categories: tag-based, attribute-based, relation-based, and ontology-based. The annotation systems within the first three categories allow minimal annotation model representation, and therefore can only enable a limited number of services that mainly focusing on basic browsing and searching functions. Knowtator [17], for example, is a attribute-based annotation environment that is well adopted by the clinical Natural Language Processing (NLP) community. Brat [1], as another example, is a web-based annotation tool for collaborative text annotation. Compared to the annotation systems in the first three categories, ontology-based annotation systems, such as Semantator, can provide semantic annotations that describe a resource with respect to a formal conceptual model. These systems allow semantic queries and reasoning. In addition to Semantator, there are other ontology-based annotation systems. Semantic-document [11] and GoNTogle [12], for example, support semantic annotation on documents with ontology classes. Compared to these systems, Semantator further supports instance relationship creation and provides reasoning capabilities. KIM [20] is a commercial software that supports manual, automatic, and semi-automatic annotation for both instances and relationships. KIM, however, does not allow users to use their own domain ontologies for annotations.

2.2. Information Extraction and Annotation Algorithms

Automatic annotation systems rely on different information extraction and annotation algorithms. Existing algorithms can be generally categorized into pattern-based systems and machine-learning-based systems. Pattern-based systems, such as PANKOW [7] and Armadillo [6], try to locate named entities by using patterns that are either manually defined or semi-automatically induced. SemTag [9] and KIM [20] use pre-defined rules to locate the information of interest. Alternatively, systems such as S-CREAM [15] and MnM [27] use machine learning and NLP-based techniques to identify named entities. Although machine-learning-based approaches do not fully rely on manually defined rules, they are usually supervised algorithms, which require certain amount of training data that need human efforts.

For the biomedical domain, there are several well-acknowledged information extraction or annotation systems. MetaMap [5], for example, is a system to map biomedical text to UMLS Metathesaurus. The clinical Text Analysis and Knowledge Extraction System (cTAKES) [22] focuses on annotating clinical narratives to standard ontologies and terminologies such as SNOMED CT and RxNorm using NLP and machine learning based approaches. The NCBO annotator [16] is a web service that helps to match biomedical text with ontology terms from one or more ontologies hosted in BioPortal (http://bioportal.bioontology.org/). Semantator provides an API for users to plug in and play state-of-the-art automatic annotation tools to connect them with domain ontologies.

3. Basic Semantic Annotation Functions

In this section, we describe the basic annotation functionalities of Semantator, including creating and removing ontology instances, managing instance relationships, and annotating relationships. We also introduce how different automatic annotation tools can be embedded in the Semantator environment.

3.1. Instance and Relationship Annotation

3.1.1. Creating and Removing Ontology Instances—To create instances, a user can highlight a piece of text and select a class from the domain ontology as demonstrated in Figure 1. By default, Semantator will save the highlighted string using *rdfs:label* to the newly created instance. Users can also add document fragments that describe instances of the same type into a "batch," and create them together. When deleting ontology instances, Semantator will first detect all instances for which this document fragment has been created, and users can then delete one or more of them as needed.

3.1.2. Managing Instance Relationships—The relationships between ontology instances are represented by proper ties in the ontology. For example, *<Event1, before, Event2>* means *Event1* happened before *Event2*. To create a relationship, a user will select the two instances (Figure 2(a)) and the corresponding property defined by the ontology. Please note that both instances involved in a relationship need to be created first before they can be related. A relationship between two instances can be easily deleted following a similar procedure as deleting an instance.

Relationships can also be annotated. Such information describes the metadata of a relationship and can be appended to the relationship using annotation in Semantator. Users can choose a piece of text and an existing instance to annotate a relationship. For example, we use *few hours* to annotate the "after" relation we just created (Figure 3).

3.2. Speeding Up Semantic Markups with Semi-automatic Annotation

In this section, we discuss the semi-automatic annotation feature of Semantator by utilizing well-adopted automatic annotation services. To demonstrate how to connect to automatic annotation services, we have connected Semantator with the NCBO annotator [16] and cTAKES [22].

The NCBO annotator provides a web service that takes user inputs (free text) and recognizes biomedical ontology terms hosted in BioPortal in the given text. The NCBO BioPortal [16] currently hosts more than 300 biomedical ontologies. When connecting with Semantator, these ontologies can be used as the annotation schema. After calling the service, Semantator will highlight all the automatically recognized entities and treat them as potential ontology instances. Users can then examine the results and retain those correctly identified instances from their perspectives. As an alternative, cTAKES can be called locally from its APIs to support semi-automatic annotation.

Figure 4(a) demonstrates the process of using the NCBO annotator. We have chosen the Medical Subject Headlines (MeSH) and PRotein Ontology (PRO). Semantator will then call the NCBO web service and find all matches from the selected ontologies. In Figure 4(b), we see that the automatically annotated instances are highlighted. Users can also choose to review each annotated instance and revise the annotation results if needed. In our example in Figure 4(b), the NCBO annotation service returned two matched concepts for "chemotherapy" from MeSH, but none from PRO. From these matched concepts, the user can further determine if they are correct matches.

4. Ontology-based Information Extraction

Embley et al [10] developed an approach to leverage ontologies for information extraction and introduced the concept of extraction ontology. Like other ontologies, an extraction ontology can specify concepts (classes), relationships, and constraints over these concepts and relationships. In addition, an extraction ontology defines a data frame for each concept that declares recognition semantics of the concept. The recognition semantics in data frames is usually represented using regular expressions. The ontology-based data recognizer matches data frames to source documents to detect any candidate instances, and then uses a set of heuristics to solve ambiguous matches.

Semantator facilitates users to create their own data frame for recognizing candidate instances of a given class. We allow users to define regular expressions by using an annotation property *csre* (customized regular expression) . For each ontology class, the user can choose to define one or more *csre* properties to capture regular expressions that can help Semantator for automatic annotation. This feature is particularly useful for recognizing numeric values (i.e., date, age, height, weight, and dose), and candidate instances with a regular pattern (i.e., address, SNP ID, and gene locus). For example, we can add the following regular expressions to define time instants or duration units respectively:

- $([0]\{0,1\}[0-9]\,|\,[1][0-9]\,|2[0-3])([:])([0-5][0-9])$
- $(\backslash\mathrm{bdays}[s]\backslash\mathrm{b}|\backslash\mathrm{byear}[s]\backslash\mathrm{b}|\backslash\mathrm{bmonth}[s]\backslash\mathrm{b}|$ $\backslash\mathrm{bweek}[s]\backslash\mathrm{b}|\backslash\mathrm{bhour}[s]\backslash\mathrm{b}|\backslash\mathrm{bhr}[s]\backslash\mathrm{b}|$ $\backslash\mathrm{bminute}[s]\backslash\mathrm{b}|\backslash\mathrm{bmin}[s]\backslash\mathrm{b}|\backslash\mathrm{bsecond}[s]\backslash\mathrm{b})$

The first regular expression is used to detect time information in 24-hour format, while the second can be utilized to recognize different time units. Figure 5(a) shows that we have selected the *cntro:TimeInstant* class, which has *csre* properties defined in the ontology.

Similar to the automatic annotation process supported by the NCBO annotator and cTAKES, the recognized candidate instances are also highlighted. As we can see in Figure 5(a), all the dates appearing in the narrative have been highlighted by Semantator. Users can choose to remove those wrongly annotated candidates, if needed, as demonstrated in Figure 5(b). Please note that because the regular expressions are attached to each specific ontology class *C*, when a user decides to create the actual instances, such instances will all be instances of *C*.

5. DIFF: Comparing Annotation Results

On many occasions, it is necessary to compare the annotation results of the same documents from different annotators. For example, there is usually more than one annotator needed to perform the annotation tasks separately, in order to create training corpus or gold standards for machine learning and NLP tools. The annotation results from these different annotators then need to be compared to reach the final annotation gold standard. Comparison is also needed when evaluating automatic annotation algorithms. In this case, the automatic annotation results need to be compared with the gold standard to measure the performance and accuracy of the automatic annotation algorithms.

To facilitate the users in the above processes, Semantator provides a DIFF function that can automatically identify the differences between annotation results and display them to users. To perform DIFF in Semantator, a user will need to load the annotated files from different annotators. Semantator will check the differences on instance annotations between two annotators when the user clicks **Start**. Finally, the differences between the two annotators are displayed in a table (consistent annotations are ignored). Figure 6(a) shows the DIFF results between two annotation files. The *Position* column indicates the position offsets (the start and end positions) of the annotated strings in the original document. The second and third columns display the URI(s) of the corresponding annotated concept(s) from different annotators. Row 1 (*Position 55.83*) shows an example where the string was annotated in *Annotation One*, but not in *Annotation Two*. Row 3 (*Position 142.158*) shows an example where the string was annotated in *Annotation Two*, but not in *Annotation One*. Row 5 (*Position 344.363*) shows an example where the string was annotated in both *Annotation Two* and *Annotation One*, but with different ontology concepts.

After getting the DIFF results, a meta review can be done to check each detected difference and select the preferred annotation. When a reviewer clicks on a specific table cell, Semantator will highlight the corresponding text in the loaded clinical narrative to help the reviewer make decisions. The meta reviewer can remove any inappropriate annotations by double-clicking on a table cell. A crossline will be drawn on top of this cell, indicating that the corresponding annotation has been removed. As Figure 6(b) shows, for each row (string), a reviewer can choose to remove one or both of the annotation results. A removed annotation can also be recovered by double-clicking the corresponding cell with a crossed annotation, if needed. When a meta annotator has finished reviewing all differences, the reviewer can then export the clean annotation, which will result in two new files for the new annotation and meta data, respectively (Figure 6(b)).

6. Semantic Web based Reasoning

Semantator connects biomedical text with Semantic Web ontologies. One advantage of putting annotation results in the Semantic Web notation is the reasoning capabilities provided by Semantic Web techniques. In this section, we illustrate some benefits of connecting Semantic Web technologies to biomedical data.

6.1. Consistency Checking for Annotated Data

OWL ontologies can define cardinality constraints, data ranges of a particular class, and disjoint classes. Using these features, we can leverage state-of-the-art semantic web reasoners and self-defined rules, if necessary, to conduct automatic consistency checking on annotated data.

Based on the cardinality constraints, we can automatically check if a particular instance has the correct number of linked components as defined. For example, a particular clinical event can only happen on one time point (e.g., have at most one time stamp). If the annotator connects the event to two time stamps, and these two time stamps are different, there would be an inconsistency warning by the system.

We can check if an instance has a value in the correct data type or within the correct data ranges. The prerequisite of using this feature is that the annotated values have been specified a data type. Currently, Semantator stores all the recognized values from the original documents using the *String* data type. Based on the particular OWL class, a normalizer could be implemented to convert a recognized string value to the appropriate data type. For example, if the system expects a numeric value for a particular class, but the annotator interpreted a string value that could not be converted to a numeric value, the system could return an inconsistency warning. In addition, we can also check if an annotated value is within the correct data range, if applicable. For example, an ontology defines that patient weight needs to be between 1lb and 500 lbs. If the annotation marked 1000 lbs as a patient weight, the system would return an inconsistency warning.

In the Semantic Web, classes can be defined as disjoint with each other, which indicates that they have no instances in common. For example, two classes, *Male* and *Female* ,are disjoint. An instance can only be declared as belonging to either of these two classes [13]. Using the automatic annotation services, however, the same piece of data could be annotated as candidate instances of disjoint classes. Take the following sentence as an example:

I was pleased to inform Mr. Smith that his PSA ***today*** is undetectable.

In this example, the NCBO annotator recognized *today* as an *Organic Chemical* with the SNOMED CT ontology. A human annotator may simply annotate it to be an instance of the *TimeInstant* class from the CNTRO ontology [26]. Assuming we have the knowledge about the disjointness between the two classes: *Organic Chemical* and *TimeInstant*, Semantator will report an inconsistency.

6.2. Automatic Classification

Two classes can also be defined to be equivalent. For example, two classes *Man* and the intersection of *Human* and *some hasGender male* are equivalent and thus any instance that is declared to be a *Man* should also be an instance of the other class. If an instance, *i*, is marked as a *Patient* (which is defined as the a subclass of *Human*) and is also connected to the instance *male* through the relation *hasGender*, the system can automatically classify *i* as an instance of class *Man*. This feature could be very useful in decision support systems for automatically detecting qualified instances based on ontology definition either by description logic or rules.

6.3. Connecting to Reasoning Tools

Since the Semantator annotated data are stored in RDF with respect to domain ontologies, we can easily connect the annotated data to other semantic web-based tools. For example, we have developed a temporal reasoning framework using OWL Description Logic and the Semantic Web Rule Language (SWRL) [25]. In one of our recent projects, we used Semantator to annotate clinical narratives. The annotated data can run with our temporal reasoning framework smoothly. Previously, we have used Knowtator as the annotation tool. Since Knowtator does not work with OWL ontologies, or output RDF files, extra efforts need to be done to convert OWL ontologies to the annotation schema compatible with Knowtator and convert the output files to RDF.

7. Evaluation

7.1. System Evaluation

Semantator can be downloaded at http://informatics.mayo.edu/CNTRO/index.php/Download_Semantator. The functionality of Semantator has been evaluated by a group of five experts: two of them are ontology and Protégé experts who were not involved with the initial implementation of Semantator[2]; the remaining three are independent of Semantator development and do not have previous backgrounds in either ontologies or Protégé. All the experts were required to evaluate the Semantator annotation functionalities based on our annotation guideline (http://informatics.mayo.edu/CNTRO/index.php/Semantator). In the evaluation, each expert needed to conduct a set of representee tasks, including loading and saving documents, instance creation and deletion, relationship management, relationship annotation, and automatic named entity recognition. The annotated results have been reviewed by the experts to ensure the system can capture their original annotation purposes.

We evaluated the usability of the system based on how easy it is for a user to complete a given task independently and if that user can repeat the same tasks (functions) after at least two weeks since the user initially used the tool. Table 1 shows the results.

For loading and saving a document, one user needs consultation to finish the tasks because the user may have confused by the Semantator *File* button with the one built in with Protégé. One user was likely not aware that OWL and RDF only support binary relationships and was trying to create a ternary relationship. A ternary relationship can actually be created by using the Semantator relationship annotation function. Annotating relationships, however, is a complex task which involves several sub-tasks. Therefore, three users could not complete this task without further help. These confusions were resolved after consultation and explanation from Semantator developers, and we have updated the annotation guideline to help users void the confusions in the future. All users are able to repeat the tasks successfully.

[2] One expert participated the improvements of the functionalities after the evaluation.

These experts were also asked to provide feedback on possible improvements on the usability and functionality of the tool. Table 2 summarizes the feedback we received and the follow-up improvements we have added accordingly. Semantator saves the annotation information in an OWL file, and the annotation meta-data (e.g., color, position offsets of the annotated strings) in an XML file. Originally, users had to choose the file to be annotated, the OWL file, and the XML file in order to load or save an annotation. After the improvement, a user now only needs to specify the original document to be annotated; the corresponding OWL file and XML file will then automatically be created and loaded. We also provided an option that allows users to browse files under the same folder, one by one, by clicking the previous or next button (Figure 7 #1). When creating an instance, Semantator originally asked users to specify color for each class. This requires a lot of clicks if the annotation involves many classes. We have updated Semantator to allow colors to be assigned automatically by the system. There are also suggestions on where to save the highlighted strings in the annotation result (OWL file). By default, they are saved as *rdfs:label* for the newly created instances. The system now also allows users to choose other properties to store the strings. For managing relationships, some users prefer to handle it directly, using Protégé functions. We have included the Protégé *Individuals* frame and the *Property assertion* frame to assist users in adding new relationships and viewing existing relationships directly. As Figure 7 #3 shows, there are two relationships associated with the instance “PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED.” New relationships can be added by clicking the plus signs in the Property assertion frame and following the Protégé instructions. Another improvement we have made is to allow users to view the corresponding text when choosing an instance. For example, if we choose the first instance in the *Relationship* tab in Figure 7 #2, the corresponding text “CARDIAC ARREST” has then been highlighted in the narrative. For automatic-named entity recognition, the evaluators reported that the Bio-Portal service sometimes return a lot of recognized strings from many source ontologies. This is quite normal, since BioPortal currently hosts more than 300 domain ontologies and there could be overlaps within these ontologies. To use the Semantator service, a user is responsible to choose the proper ontologies to be used in the annotation.

7.2. Use Case Evaluation

Semantator provides an environment where annotation of data can be conducted with respect to domain ontologies. Semantator has been adopted in a project where 239 clinical documents were manually annotated [8] with respect to a domain ontology that models late stent thrombosis and the Clinical Narrative Temporal Relation Ontology (CNTRO) [26] that models the temporal information. These documents were retrieved from the Food and Drug Administration (FDA) Manufacturing and User Facility Device Experience (MAUDE) database (http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfmaude/search.cfm). From the MAUDE database, medical device adverse-event narratives, resulting in late stent thrombosis for the years 2004 through 2010, have been included in this study. The following events were annotated within the complaint files, as they are known to commonly occurr with late stent thrombosis: initial stent implantation, follow-up stent implantation, starting and stopping point of antiplatelet therapy administration, late stent thrombosis, myocardial infarction, admission to the emergency room, and surgery.

These clinical events of interest, their time information, and the temporal relationships between the events have been annotated using Semantator. The annotation has been conducted and reviewed by two experts and any disagreements in the annotation results have been resolved after discussion. We run the annotated RDF files through our temporal relation reasoning framework [25] to further infer new temporal relationships in order to answer important time-related questions. For this use case, we focus on the following three questions:

- What was the order of events within the adverse event narrative? This question can aid in identification of event sequencing patterns.
- How long after the initial stenting procedure was antiplatelet therapy discontinued? This question can be used to assess the recommended guidelines for antiplatelet administration to prevent late stent thrombosis.
- What was the duration between discontinuation of antiplatelet therapy and stent thrombosis? This question may aid in identifying the mechanism of thrombosis formation.

The evaluation results show that the system is able to answer 96% of the questions about timeline correctly and 82% of the questions about the duration. Post-evaluation error analysis indicates that the errors were relevant either to the ontology coverage of the source information or to the reasoner capacity. Semantator can complete the annotation tasks successfully as expected. This case evaluation indicates that 1) Semantator can be adopted in real applications in clinical research, and 2) the annotated results using Semantator can be easily used in semantic web based reasoning tools for further inference.

8. Discussion

After we released Semantator, we have received much positive feedback from the community. First, Semantator provides an environment that connects clinical NLP tools with semantic web technologies. Many people find it convenient to be able to view OWL ontologies, documents to be annotated, and annotation results in the same environment. Second, the community feedback indicates that the Semantator relationships are easier to follow, as the system intuitively asks a user to identify the two instances, choose an object property, and specify the subject. Third, the DIFF function provided by Semantator can be very useful for the NLP community when evaluating the performance by comparing the results with gold standards. In addition, since Semantator is implemented as a Protégé plug-in, many annotating, querying, and browsing features can be adopted directly from Protégé. This feature is particularly convenient for those users who are already familiar with Protégé.

We have received many suggestions on how to further improve Semantator from the community. First, the current version of Semantator does not capture annotator information. In the future, it will be helpful to allow annotators to input their information at the beginning of a new annotation session. The system should capture the information of human annotators or the automatic annotation tools using OWL annotation properties to preserve the provenance of the annotation. Another drawback of Semantator is the number of files needed. Currently, Semantator saves 3 files for each annotated document: the original text file, the annotated RDF/OWL file, and a metadata XML file storing the annotation information (e.g., positions and colors) for users to reload and visualize their previous annotations. In the future, it would be helpful to store all the information using RDF with respect to domain ontologies and an ontology for annotation. Another piece of feedback is that it might be more convenient if the system could reuse the same color for the same class across different annotated documents. This might be feasible by establishing a userrepository. Whenever a user wants to use Semantator, the user could choose to log in so that all history information can be loaded; thus all the choices about colors made by this user before can automatically apply.

9. Conclusion and Future Work

This paper introduced Semantator, a semantic annotation environment for connecting biomedical narratives to semantic web technologies. Semantator has a manual annotation mode, where users can manually annotate biomedical text with respect to domain ontologies. It also provides an API through which automatic information extraction or annotation tools can be connected to the Semantator environment. In the current implementation, we have included cTAKES and the NCBO annotator for automatic named entity recognition. Users can also implement rule-based automatic recognition by adding regular expressions to a particular class or property. In addition, Semantator provides a DIFF function to automatic annotation results from two human annotators or annotation tools. This feature is particularly useful to the clinical NLP community for creating gold standard training sets or evaluating annotation results. Last but not least, the reasoning capability of Semantator could assist users in finding inconsistencies and incompleteness in their annotations, and conduct automatic classification and inference of the annotated data.

Several directions still remain for future work. First, we will further improve Semantator based on the comments we received from the community and incorporate those improvements in our next release. Second, it would be useful to calculate the inter-annotator agreement between annotations of different annotators on the DIFF mode. Furthermore, we would like to enhance Semantator with some query capability so that users can submit queries (e.g., SPARQL) to search within the annotation results. For the automatic annotation mode, automatic relation extraction (in addition to automatic instance creation) could be one interesting research question to explore in the future.

Acknowledgement

This research is partially supported by the National Center for Biomedical Ontologies (NCBO) under the NIH Grant #N01-HG04028, and the NSF under Grant #0937060 to the CRA for the CIFellows Project.

References

[1]. brat rapid annotation tool. http://brat.nlplab.org/about.html

[2]. The Protégé Ontology Editor. http://protege.stanford.edu/

[3]. Literature review on health information-seeking behaviour on the web: a health consumer and health professional perspective. European Centre for Disease Prevention and Control; 2011. Technical report

[4]. Andrews, Pierre. Ilya Zaihrayeu, and Juan Pane. A classification of semantic annotation systems. Semantic Web. 2012; 3(3):223–248.

[5]. Aronson AR, Lang FM. An overview of MetaMap: historical perspective and recent advances. Journal of the American Medical Informatics Association. 2010; 17(3):229–236. [PubMed: 20442139]

[6]. Chapman, Sam; Dingli, Alexiei; Ciravegna, Fabio. Armadillo: harvesting information for the semantic web. Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval; 2004. p. 598

[7]. Cimiano, Philipp; Handschuh, Siegfried; Staab, Steffen. Towards the self-annotating web. Proceedings of the 13th international conference on World Wide Web (WWW); 2004. p. 462-471.

[8]. Clark, K. Master's thesis. University of Minnesota; 2012. Application of a temporal reasoning framework tool in analysis of medical device adverse events.

[9]. Dill, Stephen; Eiron, Nadav; Gibson, David; Gruhl, Daniel; Guha, Ramanathan V.; Jhingran, Anant; Kanungo, Tapas; Rajagopalan, Sridhar; Tomkins, Andrew; Tomlin, John A.; Zien, Jason Y. Semtag and seeker: bootstrapping the semantic web via automated semantic annotation. Proceedings of the 12th international conference on World Wide Web (WWW); 2003. p. 178-186.

[10]. Embley, DW.; Campbell, DM.; Liddle, SW.; Smith, RD. Ontology-based extraction and structuring of information from data-rich unstructured documents. Proceedings of the 7th International Conference on Information and Knowledge Management (CIKM'98); Washington D.C.: Nov. 1998 p. 52-59.

[11]. Eriksson, Henrik. An annotation tool for semantic documents. The Semantic Web: Research and Applications, 4th European Semantic Web Conference (ESWC); 2007. p. 759-768.

[12]. Giannopoulos, Giorgos; Bikakis, Nikos; Dalamagas, Theodore; Sellis, Timos K. Gontogle: A tool for semantic annotation and search. The Semantic Web: Research and Applications, 7th Extended Semantic Web Conference (ESWC); 2010. p. 376-380.

[13]. Grosof, Benjamin N.; Horrocks, Ian; Volz, Raphael; Decker, Stefan. Description logic programs: combining logic programs with description logic. Proceedings of the Twelfth International World Wide Web Conference (WWW); 2003. p. 48-57.

[14]. Halper, Fern. Four vendor views on big data and big data analytics: IBM. 2012. http://www-01.ibm.com/software/data/bigdata/

[15]. Siegfried Handschuh and Steffen Staab. Authoring and annotation of web pages in cream. Proceedings of the 11th international conference on World Wide Web (WWW); 2002. p. 462-473.

[16]. Noy, Natalya Fridman; Shah, Nigam H.; Whetzel, Patricia L.; Dai, Benjamin; Dorf, Michael; Griffth, Nicholas; Jonquet, Clement; Rubin, Daniel L.; Storey, Margaret-Anne D.; Chute, Christopher G.; Musen, Mark A. Bioportal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research. 2009; 37(Web-Server-Issue):170–173.

[17]. Ogren, Philip V. Knowtator: A protégé plug-in for annotated corpus construction. Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL); 2006.

[18]. JOE PETRO. Natural language processing in electronic health records. 2011. http://www.kevinmd.com/blog/2011/09/natural-language-processing-electronic-health-records.html

[19]. Podichetty VK, Booher J, Whitfield M, Biscup RS. Assessment of internet use and effects among healthcare professionals: a cross sectional survey. Postgrad Med J. 2006

[20]. Popov, Borislav; Kiryakov, Atanas; Ognyanoff, Damyan; Manov, Dimitar; Kirilov, Angel. Kim a semantic platform for information extraction and retrieval. Nat. Lang. Eng. Sep.2004 10:375–392.

[21]. The RDF vocabulary. http://www.w3.org/1999/02/22-rdf-syntax-ns

[22]. Savova, Guergana K.; Masanz, James J.; Ogren, Philip V.; Zheng, Jiaping; Sohn, Sunghwan; Kipper-Schuler, Karin C.; Chute, Christopher G. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. Journal of the American Medical Informatics Association. 2010; 17(5):507–513. [PubMed: 20819853]

[23]. Song D, Chute CG, Tao C. Semantator: annotating clinical narratives with semantic web ontologies. AMIA Summits Transl Sci Proc. 2012:2012.

[24]. SPARQL Query Language for RDF. www.w3.org/TR/rdf-sparql-query/

[25]. Tao, Cui; Solbrig, Harold R.; Sharma, Deepak K.; Wei, Wei-Qi; Savova, Guergana K.; Chute, Christopher G. Time-oriented question answering from clinical narratives using semantic-web techniques. 9th International Semantic Web Conference (ISWC); 2010. p. 241-256.

[26]. Tao, Cui; Wei, Wei-Qi; Solbrig, Harold R.; Savova, Guergana; Chute, Christopher G. CNTRO: A semantic web ontology for temporal relation inferencing in clinical narratives. American Medical Informatics Assocication Annual Symposium (AMIA); 2010. p. 787-91.

[27]. Vargas-Vera, Maria; Motta, Enrico; Domingue, John; Lanzoni, Mattia; Stutt, Arthur; Ciravegna, Fabio. Mnm: Ontology driven semi-automatic and automatic support for semantic markup. 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW); 2002. p. 379-391.

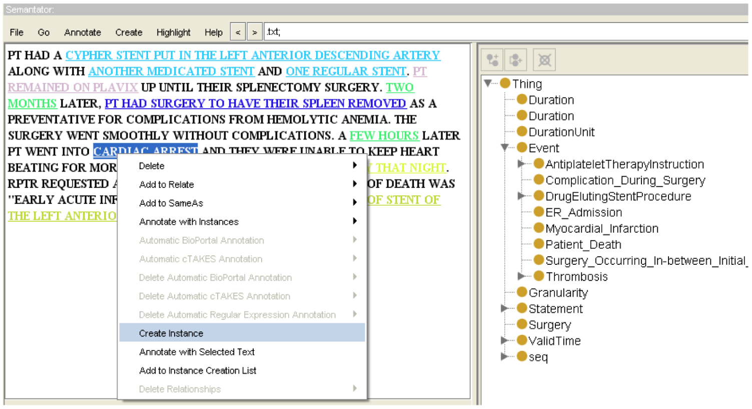
> We introduce Semantator: a Semantic Web based Semantic Annotation environment.

> Semantator converts biomedical text to linked data with a formal semantics.

> It provides an environment where annotation can be viewed in its original context.

> Evaluation indicated that it can perform the annotation tasks as designed.

> Semantator is publically accessible online.

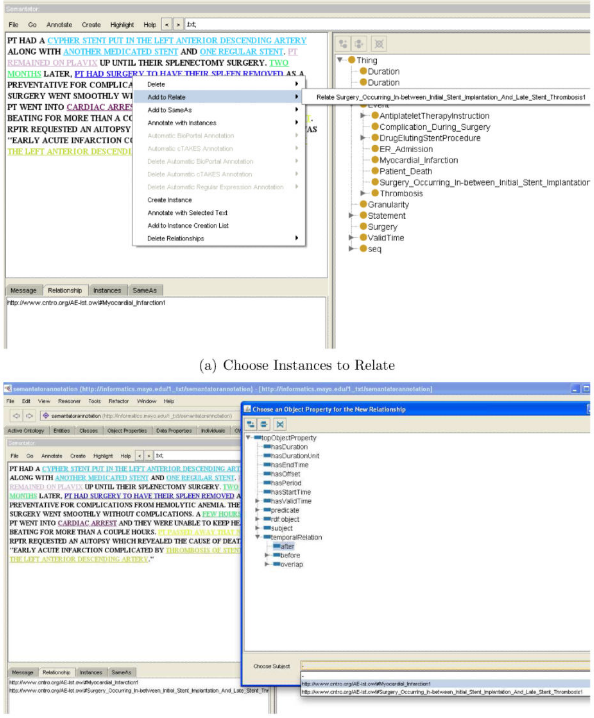


PT HAD A CYPHER STENT PUT IN THE LEFT ANTERIOR DESCENDING ARTERY
ALONG WITH ANOTHER MEDICATED STENT AND ONE REGULAR STENT. PT
REMAINED ON PLAVIX UP UNTIL THEIR SPLENECTOMY SURGERY. TWO
MONTHS LATER, PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED AS A
PREVENTATIVE FOR COMPLICATIONS FROM HEMOLYTIC ANEMIA. THE
SURGERY WENT SMOOTHLY WITHOUT COMPLICATIONS. A FEW HOURS LATER
PT WENT INTO CARDIAC ARREST AND THEY WERE UNABLE TO KEEP HEART
BEATING FOR MOR Y THAT NIGHT.
RPTR REQUESTED A OF DEATH WAS
"EARLY ACUTE INF OF STENT OF
THE LEFT ANTERIO

Semantator:
File
Go
Annotate
Create
Highlight
Help
<
>
.txt;
Delete
Add to Relate
Add to SameAs
Annotate with Instances
Automatic BioPortal Annotation
Automatic cTAKES Annotation
Delete Automatic BioPortal Annotation
Delete Automatic cTAKES Annotation
Delete Automatic Regular Expression Annotation
Create Instance
Annotate with Selected Text
Add to Instance Creation List
Delete Relationships
Thing
Duration
Duration
DurationUnit
Event
AntiplateletTherapyInstruction
Complication_During_Surgery
DrugElutingStentProcedure
ER_Admission
Myocardial_Infarction
Patient_Death
Surgery_Occurring_In-between_Initial_
Thrombosis
Granularity
Statement
Surgery
ValidTime
seq

Figure 1.

An Example of Instance Creation



PT HAD A CYPHER STENT PUT IN THE LEFT ANTERIOR DESCENDING ARTERY
ALONG WITH ANOTHER MEDICATED STENT AND ONE REGULAR STENT. PT
REMAINED ON PLAVIX UP UNTIL THEIR SPLENECTOMY SURGERY. TWO
MONTHS LATER, PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED AS A
PREVENTATIVE FOR COMPLICA
SURGERY WENT SMOOTHLY WI
PT WENT INTO CARDIAC ARRES
BEATING FOR MORE THAN A CO
RPTR REQUESTED AN AUTOPSY
"EARLY ACUTE INFARCTION CO
THE LEFT ANTERIOR DESCENDI

http://www.cntro.org/AE-lst.owl#Myocardial_Infarction1

Semantator
File
Go
Annotate
Create
Highlight
Help
< >
.txt;
Delete
Add to Relate
Relate Surgery_Occurring_In-between_Initial_Stent_Implantation_And_Late_Stent_Thrombosis1
Add to SameAs
Annotate with Instances
Automatic BioPortal Annotation
Automatic cTAKES Annotation
Delete Automatic BioPortal Annotation
Delete Automatic cTAKES Annotation
Delete Automatic Regular Expression Annotation
Create Instance
Annotate with Selected Text
Add to Instance Creation List
Delete Relationships
Thing
Duration
Duration
Event
AntiplateletTherapyInstruction
Complication_During_Surgery
DrugElutingStentProcedure
ER_Admission
Myocardial_Infarction
Patient_Death
Surgery_Occurring_In-between_Initial_Stent_Implantation
Thrombosis
Granularity
Statement
Surgery
ValidTime
seq
Message
Relationship
Instances
SameAs

(a) Choose Instances to Relate

PT HAD A CYPHER STENT PUT IN THE LEFT ANTERIOR DESCENDING ART
ALONG WITH ANOTHER MEDICATED STENT AND ONE REGULAR STENT.
REMAINED ON PLAVIX UP UNTIL THEIR SPLENECTOMY SURGERY. TWO
MONTHS LATER, PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED A
PREVENTATIVE FOR COMPLICATIONS FROM HEMOLYTIC ANEMIA. THE
SURGERY WENT SMOOTHLY WITHOUT COMPLICATIONS. A FEW HOURS
PT WENT INTO CARDIAC ARREST AND THEY WERE UNABLE TO KEEP HE
BEATING FOR MORE THAN A COUPLE HOURS. PT PASSED AWAY THAT N
RPTR REQUESTED AN AUTOPSY WHICH REVEALED THE CAUSE OF DEAT
"EARLY ACUTE INFARCTION COMPLICATED BY THROMBOSIS OF STEN
THE LEFT ANTERIOR DESCENDING ARTERY."

http://www.cntro.org/AE-lst.owl#Myocardial_Infarction1
http://www.cntro.org/AE-lst.owl#Surgery_Occurring_In-between_Initial_Stent_Implantation_And_Late_Stent_Thr

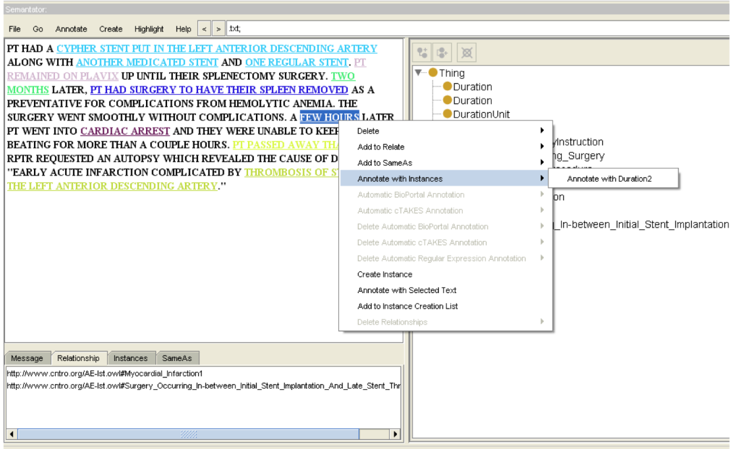
Choose Subject: -

-
http://www.cntro.org/AE-lst.owl#Myocardial_Infarction1
http://www.cntro.org/AE-lst.owl#Surgery_Occurring_In-between_Initial_Stent_Implantation_And_Late_Stent_Thrombosis1

(b) Choose Subject

Figure 2.

Connecting Instances with Ontological Properties

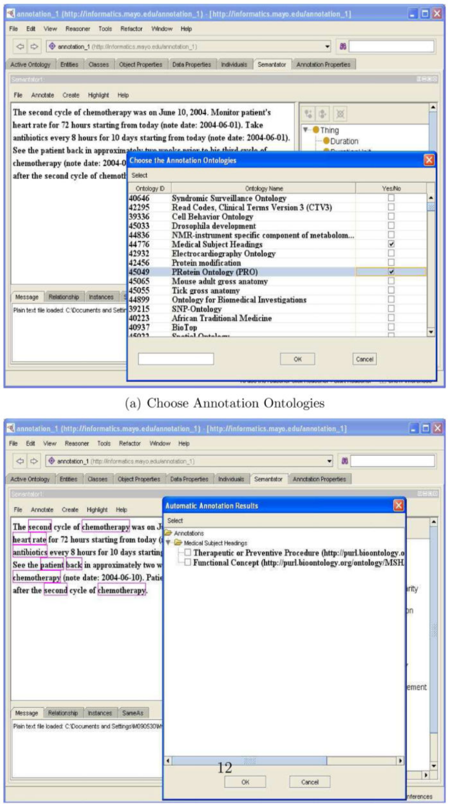

Semantator:
File
Go
Annotate
Create
Highlight
Help
<
>
.txt;
Thing
Duration
Duration
DurationUnit
yInstruction
ng_Surgery
on
_In-between_Initial_Stent_Implantation
Delete
Add to Relate
Add to SameAs
Annotate with Instances
Annotate with Duration2
Automatic BioPortal Annotation
Automatic cTAKES Annotation
Delete Automatic BioPortal Annotation
Delete Automatic cTAKES Annotation
Delete Automatic Regular Expression Annotation
Create Instance
Annotate with Selected Text
Add to Instance Creation List
Delete Relationships
Message
Relationship
Instances
SameAs

PT HAD A CYPHER STENT PUT IN THE LEFT ANTERIOR DESCENDING ARTERY
ALONG WITH ANOTHER MEDICATED STENT AND ONE REGULAR STENT. PT
REMAINED ON PLAVIX UP UNTIL THEIR SPLENECTOMY SURGERY. TWO
MONTHS LATER, PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED AS A
PREVENTATIVE FOR COMPLICATIONS FROM HEMOLYTIC ANEMIA. THE
SURGERY WENT SMOOTHLY WITHOUT COMPLICATIONS. A FEW HOURS LATER
PT WENT INTO CARDIAC ARREST AND THEY WERE UNABLE TO KEEP
BEATING FOR MORE THAN A COUPLE HOURS. PT PASSED AWAY THA
RPTR REQUESTED AN AUTOPSY WHICH REVEALED THE CAUSE OF D
"EARLY ACUTE INFARCTION COMPLICATED BY THROMBOSIS OF ST
THE LEFT ANTERIOR DESCENDING ARTERY."

```
http://www.cntro.org/AE-lst.owl#Myocardial_Infarction1
http://www.cntro.org/AE-lst.owl#Surgery_Occurring_In-between_Initial_Stent_Implantation_And_Late_Stent_Thr
```

Figure 3.

Annotating Instance Relationships



annotation_1 (http://informatics.mayo.edu/annotation_1) - [http://informatics.mayo.edu/annotation_1]
File Edit View Reasoner Tools Refactor Window Help
annotation_1 (http://informatics.mayo.edu/annotation_1)
Active Ontology Entities Classes Object Properties Data Properties Individuals Semantator Annotation Properties
Semantator1
File Annotate Create Highlight Help
Thing
Duration
Message Relationship Instances
Choose the Annotation Ontologies
Select
OK Cancel

The second cycle of chemotherapy was on June 10, 2004. Monitor patient's heart rate for 72 hours starting from today (note date: 2004-06-01). Take antibiotics every 8 hours for 10 days starting from today (note date: 2004-06-01). See the patient back in approxim chemotherapy (note date: 2004-0 after the second cycle of chemoth

Plain text file loaded: C:\Documents and Settir

Ontology ID	Ontology Name	Yes/No
40646	Syndromic Surveillance Ontology	□
42295	Read Codes, Clinical Terms Version 3 (CTV3)	□
39336	Cell Behavior Ontology	□
45033	Drosophila development	□
44836	NMR-instrument specific component of metabolom...	□
44776	Medical Subject Headings	☑
42932	Electrocardiography Ontology	□
42456	Protein modification	□
45049	PRotein Ontology (PRO)	☑
45065	Mouse adult gross anatomy	□
45055	Tick gross anatomy	□
44899	Ontology for Biomedical Investigations	□
39215	SNP-Ontology	□
40223	African Traditional Medicine	□
40937	BioTop	□

(a) Choose Annotation Ontologies

annotation_1 (http://informatics.mayo.edu/annotation_1) - [http://informatics.mayo.edu/annotation_1]
File Edit View Reasoner Tools Refactor Window Help
annotation_1 (http://informatics.mayo.edu/annotation_1)
Active Ontology Entities Classes Object Properties Data Properties Individuals Semantator Annotation Properties
Semantator1
File Annotate Create Highlight Help
Message Relationship Instances SameAs
Automatic Annotation Results
Select
Annotations
Medical Subject Headings
12
OK Cancel
rity
on
ement
nferences

The second cycle of chemotherapy was on J heart rate for 72 hours starting from today (antibiotics every 8 hours for 10 days starting See the patient back in approximately two w chemotherapy (note date: 2004-06-10). Patie after the second cycle of chemotherapy.

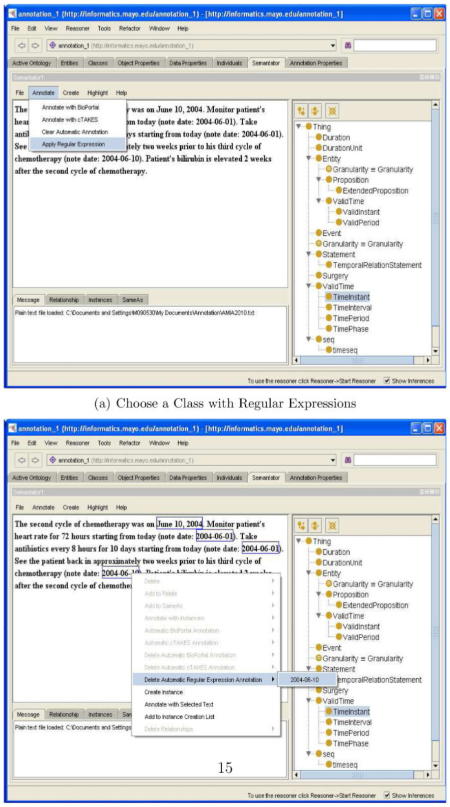
Plain text file loaded: C:\Documents and Settings\M090530\M

- □ Therapeutic or Preventive Procedure (http://purl.bioontology.o
- □ Functional Concept (http://purl.bioontology.org/ontology/MSH

(b) Retain Correctly Identified Instances

Figure 4.

Semi-automatic Annotation with BioPortal Web Services



annotation_1 (http://informatics.mayo.edu/annotation_1) - [http://informatics.mayo.edu/annotation_1]
File Edit View Reasoner Tools Refactor Window Help
annotation_1 (http://informatics.mayo.edu/annotation_1)
Active Ontology Entities Classes Object Properties Data Properties Individuals Semantator Annotation Properties
Semantator1
File Annotate Create Highlight Help
Annotate with BioPortal
Annotate with cTAKES
Clear Automatic Annotation
Apply Regular Expression
Thing
Duration
DurationUnit
Entity
Granularity ≡ Granularity
Proposition
ExtendedProposition
ValidTime
ValidInstant
ValidPeriod
Event
Granularity ≡ Granularity
Statement
TemporalRelationStatement
Surgery
ValidTime
TimeInstant
TimeInterval
TimePeriod
TimePhase
seq
timeseq
Message Relationship Instances SameAs
To use the reasoner click Reasoner->Start Reasoner
Show Inferences

The y was on June 10, 2004. Monitor patient's hear m today (note date: 2004-06-01). Take antil ys starting from today (note date: 2004-06-01). See tely two weeks prior to his third cycle of chemotherapy (note date: 2004-06-10). Patient's bilirubin is elevated 2 weeks after the second cycle of chemotherapy.

Plain text file loaded: C:\Documents and Settings\M090530\My Documents\Annotation\AMIA2010.txt

(a) Choose a Class with Regular Expressions

Delete
Add to Relate
Add to SameAs
Annotate with Instances
Automatic BioPortal Annotation
Automatic cTAKES Annotation
Delete Automatic BioPortal Annotation
Delete Automatic cTAKES Annotation
Delete Automatic Regular Expression Annotation
2004-06-10
Create Instance
Annotate with Selected Text
Add to Instance Creation List
Delete Relationships
15

The second cycle of chemotherapy was on June 10, 2004. Monitor patient's heart rate for 72 hours starting from today (note date: 2004-06-01). Take antibiotics every 8 hours for 10 days starting from today (note date: 2004-06-01). See the patient back in approximately two weeks prior to his third cycle of chemotherapy (note date: 2004-06-

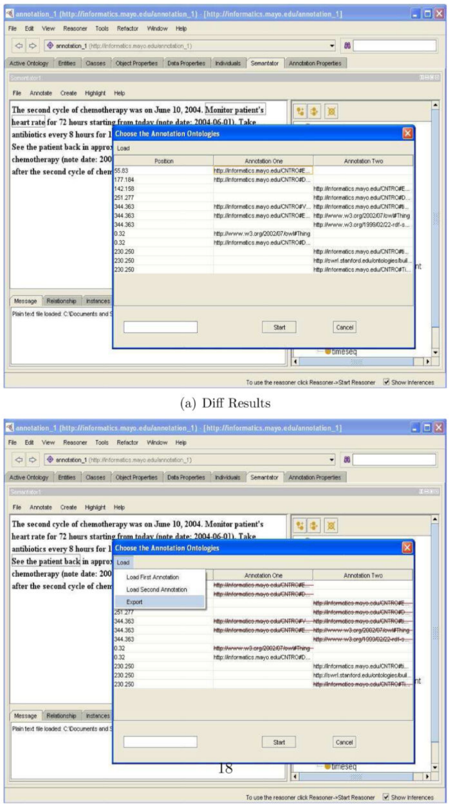
after the second cycle of chemothe

Plain text file loaded: C:\Documents and Settings

(b) Generate Instances Recognized by Regular Expressions

Figure 5.

Regular Expression based Semi-automatic Annotation



The second cycle of chemotherapy was on June 10, 2004. Monitor patient's heart rate for 72 hours starting from today (note date: 2004-06-01). Take antibiotics every 8 hours for 1 See the patient back in approx chemotherapy (note date: 200 after the second cycle of chem

Position	Annotation One	Annotation Two
55.83	http://informatics.mayo.edu/CNTRO#E...	
177.184	http://informatics.mayo.edu/CNTRO#D...	
142.158		http://informatics.mayo.edu/CNTRO#E...
251.277		http://informatics.mayo.edu/CNTRO#D...
344.363	http://informatics.mayo.edu/CNTRO#V...	http://informatics.mayo.edu/CNTRO#ti...
344.363	http://informatics.mayo.edu/CNTRO#E...	http://www.w3.org/2002/07/owl#Thing
344.363		http://www.w3.org/1999/02/22-rdf-s...
0.32	http://www.w3.org/2002/07/owl#Thing	
0.32	http://informatics.mayo.edu/CNTRO#D...	
230.250		http://informatics.mayo.edu/CNTRO#ti...
230.250		http://swrl.stanford.edu/ontologies/buil...
230.250		http://informatics.mayo.edu/CNTRO#Ti...

Plain text file loaded: C:\Documents and S

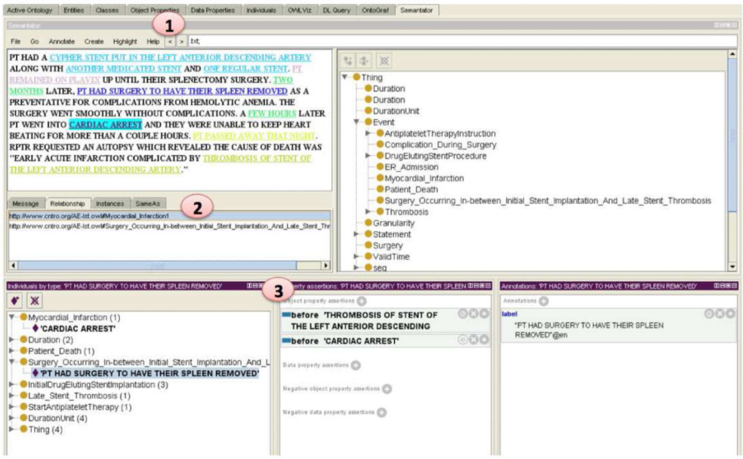
(a) Diff Results

Position	Annotation One	Annotation Two
	~~http://informatics.mayo.edu/CNTRO#E...~~	
	~~http://informatics.mayo.edu/CNTRO#D...~~	
		~~http://informatics.mayo.edu/CNTRO#E...~~
251.277		~~http://informatics.mayo.edu/CNTRO#D...~~
344.363	~~http://informatics.mayo.edu/CNTRO#V...~~	~~http://informatics.mayo.edu/CNTRO#ti...~~
344.363	~~http://informatics.mayo.edu/CNTRO#E...~~	~~http://www.w3.org/2002/07/owl#Thing~~
344.363		~~http://www.w3.org/1999/02/22-rdf-s...~~
0.32	~~http://www.w3.org/2002/07/owl#Thing~~	
0.32	http://informatics.mayo.edu/CNTRO#D...	
230.250		http://informatics.mayo.edu/CNTRO#ti...
230.250		http://swrl.stanford.edu/ontologies/buil...
230.250		~~http://informatics.mayo.edu/CNTRO#Ti...~~

(b) Export Diff Results

Figure 6.

Check the Differences between Two Annotations


Active Ontology
Entities
Classes
Object Properties
Data Properties
Individuals
OWLViz
DL Query
OntoGraf
Semantator
1
File
Go
Annotate
Create
Highlight
Help
<
>
.txt;
Message
Relationship
Instances
SameAs
2
Thing
Duration
Duration
DurationUnit
Event
AntiplateletTherapyInstruction
Complication_During_Surgery
DrugElutingStentProcedure
ER_Admission
Myocardial_Infarction
Patient_Death
Surgery_Occurring_In-between_Initial_Stent_Implantation_And_Late_Stent_Thrombosis
Thrombosis
Granularity
Statement
Surgery
ValidTime
seq
3
Individuals by type: 'PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED'
Myocardial_Infarction (1)
Duration (2)
Patient_Death (1)
Surgery_Occurring_In-between_Initial_Stent_Implantation_And_L
InitialDrugElutingStentImplantation (3)
Late_Stent_Thrombosis (1)
StartAntiplateletTherapy (1)
DurationUnit (4)
Thing (4)
Object property assertions
Data property assertions
Negative object property assertions
Negative data property assertions
Annotations: 'PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED'
Annotations
label

PT HAD A CYPHER STENT PUT IN THE LEFT ANTERIOR DESCENDING ARTERY ALONG WITH ANOTHER MEDICATED STENT AND ONE REGULAR STENT. PT REMAINED ON PLAVIX UP UNTIL THEIR SPLENECTOMY SURGERY. TWO MONTHS LATER, PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED AS A PREVENTATIVE FOR COMPLICATIONS FROM HEMOLYTIC ANEMIA. THE SURGERY WENT SMOOTHLY WITHOUT COMPLICATIONS. A FEW HOURS LATER PT WENT INTO CARDIAC ARREST AND THEY WERE UNABLE TO KEEP HEART BEATING FOR MORE THAN A COUPLE HOURS. PT PASSED AWAY THAT NIGHT. RPTR REQUESTED AN AUTOPSY WHICH REVEALED THE CAUSE OF DEATH WAS "EARLY ACUTE INFARCTION COMPLICATED BY THROMBOSIS OF STENT OF THE LEFT ANTERIOR DESCENDING ARTERY."

http://www.cntro.org/AE-lst.owl#Myocardial_Infarction1
http://www.cntro.org/AE-lst.owl#Surgery_Occurring_In-between_Initial_Stent_Implantation_And_Late_Stent_Thr

'CARDIAC ARREST'
'PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED'

...erty assertions: 'PT HAD SURGERY TO HAVE THEIR SPLEEN

before 'THROMBOSIS OF STENT OF THE LEFT ANTERIOR DESCENDING
before 'CARDIAC ARREST'

"PT HAD SURGERY TO HAVE THEIR SPLEEN REMOVED"@en

Figure 7.

Semantator User Interface

Table 1

Usability Evaluation on Representative Tasks

Function	Needs consultation to complete function	Ability to repeat function
1. Load Document	yes: 1; no: 4	yes: 5
2. Create Instance	yes: 0; no: 5	yes: 5
3. Delete Instance	yes: 0; no: 5	yes: 5
4. Create Relation	yes: 1; no: 4	yes: 5
5. Delete Relation	yes: 0; no: 5	yes: 5
6. Annotate Relation	yes: 3; no: 2	yes: 5
7. Save Annotation	yes: 1; no: 4	yes: 5
8. Automatic Named Entity Recognition	yes: 0; no: 5	yes: 5

Table 2

Feedback Summary Received from the Evaluators and the Follow Up Actions to Improve the System Accordingly

Function	Feedback	Response and Improvement
1. Load and save document	• Open an annotated file without too many clicks • Open a set of documents at once	• Open an annotated file according to the file name convention • Provide an option to allow users to browse all the files in the same folder one by one by clicking the previous or next button
2. Create instances	• Too many clicks for choosing the mark-up colors for the instances • The annotated text does not necessarily need to be saved as *rdfs:label* of the created instance	• Assign the colors automatically • Allow users to choose a property to store the annotated text
3. Delete instance	NA	NA
4. Manage relationship	• A user who is familiar with Protégé may want to create or delete the relationships between instances directly using Protégé functions • Sometimes it is difficult to know the instance content (the corresponding text in the narrative) by looking at the URIs list in the Relationship tab	• Added Protégé Individuals and Property Assertion frames to the Semantator Tab, to allow the relations to be created using these frames directly (Figure 7 #3) • When an instance in the Relationship tab is chosen, the corresponding text in the narrative will be highlighted (Figure 7 #2)
5. Annotate relationship	How to delete a relation annotation	Currently, the annotation can either be deleted using the Protégé Property assertion frame or by deleting the relation itself using Semantator and recreating the relationship without annotation
6. Automatic named entity recognition	BioPortal service sometimes returns a lot of recognized strings from many source ontologies. Many of them need to be removed from the annotation results	The assumption of using this service is that the user can choose the proper ontologies for the annotation