NATURAL LANGUAGE QUESTIONS FOR SEMANTIC WEB-BASED NEWS AGGREGATOR

Quang-Minh Nguyen¹ and Tuan-Dung Cao²*

¹School of Electronics and Telecommunications, Hanoi University of Science and Technology, Vietnam
²School of Information and Communication Technology, Hanoi University of Science and Technology, Vietnam
*dungct@soict.hust.edu.vn

Abstract— The impressive growth of the Web in the last decade has made it becomes one of the most popular news sources. Web-users nowadays can access an unlimited amount of news items but only a subset of them meets their interest. The existing news systems on Web reveal more and more limitations in information search because news items are presented only for human consumption. This paper presents BKSport, a sports news aggregation system that provides users with the ability to find relevant news articles through queries that are formulated in natural language. Our contribution consists of a system development approach, which is totally based on Semantic Web technologies. First, semantic annotations about news item are created using an ontology and knowledgebase for sports domain. Additionally, we propose a method to transform natural language questions into SPARQL queries to carry out semantic search on these semantic annotations. Last, the system has been positively evaluated with respect to precision, based on a set of pre-defined questions belonging to various categories.

Keywords— Natural language interface, semantic web, question answering, web news aggregator, information retrieval, SPARQL

1. INTRODUCTION

Nowadays, sport is one of the topics that attract newsreaders the most. In parallel, the Web is gradually becoming more and more popular as a source of sports news thanks to its high availability and convenient accessibility. Contents about sports are updated continually on news websites from heterogeneous sources. Subsequently, a massive number of daily news articles published on the website of different news agencies and television channels confuses readers when they want to choose the one that meets their interests. Therefore, providing accurate and friendly information search feature is always among the most important objectives in the development of Web-based news systems [1]. Additionally, on the editor side, it is more difficult and time-consuming to arrange, aggregate and recommend news items. Many news systems are built on the basis of a content management system and use keyword search engines to filter news items [14]. However, this traditional searching method lacks the power of understanding the meaning of user queries.

The underlying cause of the aforementioned problems comes from the syntactic model of representing and storing information and it is crucial to make computer aware of more activities in a web news system, especially information retrieval and manipulation.
Semantic Web technologies promise to overpass these barriers by providing machine-understandable semantics of data and enabling automated processing of information on the web. In this paper, we present BKSport, a semantic web-based sports news aggregation system that can respond to the need for advanced knowledge management and information retrieval of the news industry. Our approach is to define models for the representation of news articles and users searching queries using a domain-specific ontology. First, semantic annotations about news item are created using an ontology and knowledgebase for the sports domain. These metadata, as known as semantic annotations, are the basis for implementing semantic-based functionalities including news searching and recommending. Second, we propose a method to transform natural language questions into SPARQL queries to carry out semantic search on these semantic annotations. By applying the proposed method, BKSport can understand complex questions such as "Which team defeated Arsenal this season?" or "Who transferred to Real Madrid this year?" and returns both precise and suitable articles relevant to readers’ intention and even provides useful supplementary information.

The rest of this paper is organized as follows. We begin with the literature review on Semantic Web based systems and particularly the question answering systems in Section 2. Section 3 elaborates on the approach for developing the news aggregation system and the overall system architecture. The construction of the ontology and the knowledge base on sports is discussed in Section 4. Section 5 briefly introduces the automatic semantic annotation sub-system while Section 6 describes the method for translating a natural language question in the sports domain to a SPARQL query. The experimental results of the proposed method are presented in Section 7, followed by the conclusion.

2. RELATED WORKS

The positive aspects of applying Semantic Web such as scalability and the reasoning capabilities of inference engines; smooth deployment are demonstrated in certain works [9][18]. Many semantic Web applications have been developed to search, bookmark information and to expand the search based on the terminology of ontology [17]. The semantic web helps improve the quality of information processing with intelligent search based on semantic reasoning. The Esperonto portal [10] is the foundation for the EU Esperonto project, which exposes three levels of access: keyword-based search, ontology browsing, and getting all instances of that concept and its sub-concepts at each step of the browse. The strength of Esperonto portal is the ontology management utility, however, its user interface is not friendly and the ability to process and access information is still limited.

SPARQL queries are essential for implementing the semantic search in many systems; however, this language is not suitable to normal users. Certain works enhanced user-friendliness by providing graphical user interface based on ontology to formulate SPARQL queries [6] [16], but they still required users to engage in certain manipulation and basically understand ontology.

There have been some efforts to improve interactions in Question-Answering systems by way of using the natural language. PANTO [15] is a Portable nAtural language interface for ontologies allowing users to represent “information needs” using natural language without knowledge of RDF, OWL syntax, query languages, or vocabularies of the ontologies. It uses WordNet and String metric algorithm to map words in the user’s query to entities in the ontology (concepts, entities, relations). It uses the Stanford Parser to analyze the question into a parse tree, and then extracts nominal phrases to constitute the intermediate representation, named QueryTriples Form. However, the system still has limitations in processing negative questions and is unable to deal with the question of quantity.
QuestIO [7] (Question-based Interface to Ontologies) is a tool which uses natural language to serve queries of large knowledge stores in ontology. The drawback of this tool stems from the detection of relations of the question as it is based on rules and not on syntax analysis of questions at a deep level. Therefore, it cannot deal with questions with complex semantic. The test was carried out on data set of 22 questions from the GATE user mailing list, where users enquired about various GATE steps and plugins. The precision obtained was 71.88%.

ORAKEL [5] is a natural language interface which accepts questions and returns answers on the basis of a given ontology. This task is performed by virtue of the Query Interpreter which interprets the input question and transforms it to a first-order logic representation. The Query Converter, then, transforms the logical representation of the question into a SPARQL query. It requires two sets: domain-specific and domain-independent vocabularies. This system can operate with just a given ontology and a set of domain-specific vocabulary. This system can only work with the wh-question type and not the yes/no question type.

Certain works based on controlled natural languages, such as GiNSENG [2], typically consider an unambiguous restricted subset of natural language that can be translated directly into SPARQL. Squall2Sparql [8] typically consider SQUALL, an unambiguous restricted subset of natural language that can be translated directly into SPARQL. It achieves high performance but manual annotations of the URIs are required.

The above studies show the importance of semantic search in the form of natural language question. Many among these systems such as PANTO and QuestIO focus on general domains rather than specific ones. However, porting Natural Language Interfaces to other domains [3][5] is not an easy task as even though the approaches are not based on any controlled natural language. Up to now, rare studies report high performance when applying for sport domain. While the question types are identified by the above systems [5] usually having simple structures, our work aims to handle more complex forms of questions such as comparative question, superlative question, questions with multiple subjects and objects, question with abnormal grammar structure, questions with temporal constraint.

3. SEMANTIC WEB APPROACH FOR NEWS AGGREGATION AND SYSTEM ARCHITECTURE

At this point in time, conventional web technologies are used for building news systems. However, the support for essential information processing tasks like access, search, visualization and recommendation is not precise as expected. As mentioned above, Semantic Web technologies bring many benefits when applied to information systems and software in different domains. However, there has not been much study on the application of the Semantic Web in the domain of sports in general and sports news in particular. Believing that these technologies can considerably defeat the shortcomings of current news system, we propose to use them as the key to the construction of our news aggregation system, BKSport. The main idea is to insert machine-readable data, as known as semantic annotation, to every news item. The metadata are represented using consensus vocabulary defined in an Ontology, which is the basis for implementing semantic-based functionalities including news searching and recommending. Other sayings, BKSport relies on a semantic model for news article representation instead of the syntactic one.

An overview of the architecture behind BKSport is given in Figure 1. The system is designed with component-based architecture to support semantic-based indexing and retrieval over large news articles collections on Sport domain.
The Crawler component collects news items from popular sports news websites such as BBCSport, ESPN, SkySport. Collected news items in HTML format are parsed and the necessary information such as title, description, content, and so on, is extracted and stored in the news database. For each news item, the semantic annotation system recognizes and extracts semantic annotation based on ontology and knowledge base. Semantic information is transformed into metadata in the form of n-quad and then saved to the annotation base. Articles stored in the news database and the supplementary semantic annotation can be loaded into the Web-based User Interface component to be displayed to readers. The Recommender component is responsible for analyzing the article that users are reading and suggest related news items based on semantic similarity. Besides, the User Interface also provides GUI to perform the semantic search across news data using question-answering in natural language. First, a reader’s question in natural language is transformed into a SPARQL query with the help of the query transformer. The Query Answering takes SPARQL queries as input and presents queries to the annotation base and the news database for obtaining news and related information to display to readers.

4. SPORT ONTOLOGY AND KNOWLEDGE BASE CONSTRUCTION

Providing a controlled vocabulary for the representation of the shared background knowledge for a community, the ontology is at the center and plays evidently an important role and has a big impact on main components of BKSPORT. Below are the advantages the usage of ontology contributes to the system:

- To enable heterogeneous sport news articles integration from multiple sources. The shared vocabulary and meanings for terms with respect to other terms in the domain of sports, diverse data can be standardized and matched.
- To remove ambiguity and imprecision in search results by applying semantic search and reasoning on sport news articles.

We follow Gruber’s principles in the construction of BKSPORT ontology to assure the clarity, consistency. As the BKSPORT Ontology is used in most important components like Automatic Semantic Annotation, Semantic Search and Recommendation, the high coverage of vocabulary is the essential requirement. The terms in the ontology must be sufficient to be able to express the important semantics inside news articles or user questions for example: “Rashford is a forward and plays for Manchester United, head coach of which is Jose Mourinho” or “Real Madrid defeats Barcelona with score 2-0”.

The BKSPORT Ontology is designed having three layers corresponding to different levels of abstraction of terminology. On the top layer, abstract concepts and properties of PROTON ontology are inherited to reuse the KIM [13] platform for the task of named entity detection. In the middle layer, we selected and reused useful parts of BBC sport...
Ontology, which allow representing competitive sports events. The bottom layer is dedicated to the specific vocabularies which are indispensable for the representation of important sports figures, activities and events. These concepts and properties are identified through our thorough analysis of news articles and frequently searched questions on sports news websites. Figure 2 visually sketches the partial view of BKSport ontology.

![Diagram of BKSport ontology](image)

Fig. 2 Partial view of BKSport ontology

Named entity recognition is an essential task, of which the results affect the accuracy and the effectiveness of important components such as Semantic Annotation or Query Transformer. A large knowledge base on sports domain has been building in the aim of improving the overall system performance. Web Scrappers were implemented to collect data about famous sports figures, organizations, and places from sports websites and online databases. The extracted sport data are matched with BKSport ontology to be transformed to metadata in RDF format and enriched with their alias or synonyms in the Knowledge Base Enrichment module. As BKSport Ontology is conforming to PROTON Ontology, the resulting knowledgebase can be integrated into the knowledge base of KIM.

5. AUTOMATIC SEMANTIC ANNOTATION OF NEWS ARTICLES

The components Semantic Search and News Recommender cannot show their outstanding results without the availability of good quality semantic annotations. In the scope of research, the semantic elements of news items should be annotated in the following two types:

1. Properties of a news item (i.e., URL, created time, subject, and so on), entities mentioned in news (i.e., player, team, league, and so on), and properties of these entities, such as role of a player, a team a player is playing for, league the team participates in, and so on. For this type, semantic annotations are represented using RDF. For example:

   `<owl:NamedIndividual rdf:about="http://bk.sport.owl#jonathan-viera">`
   `
   <bksport:playFor rdf:resource="http://bk.sport.owl#ud-las-palmas"/>`
   `
   <protons:generatedBy rdf:resource="http://bk.sport.owl">`
   `
   <protons:hasAlias>Jonathan Viera</protons:hasAlias>`


2. Activities mentioned in a news item, i.e., scorers, player transfer, match between two teams, and so on). In this case, quadruple is proposed as a representation form for semantic annotations because they are events taking place in the context of news. For example:

```
PREFIX bksport: <http://bk.sport.owl#>
<brksp#wayne_rooney> <bksport#playFor> <bksport#manchester_united_fc>
<bksport#namedgraph>
```

The main drawback of manual annotation methods is the nonconformity with large data scales of daily news updated in an aggregation system. To implement the Semantic Annotation System, we proposed methods for generating semantic annotations of sport news items which concentrate both on the accuracy and the diversity of detected semantics. To achieve this goal, efforts in improvement of named entities detection and semantic triples extraction has been made thanks to the application of Sport ontology and knowledge base.

KIM [13] is a platform that provides services for automatic text semantic annotation in public domains. To identify named entities in news items as instances in ontology, we proposed to extend KIM for working with a sports domain by designing BKSPORT ontology and knowledgebase compatible and integratable with the PROTON ones. By mapping using subClassOf predicate, the specialized concepts and properties in the BKSPORT ontology can mask the abstract terms in Proton Ontology in the recognition process.

Besides reusing KIM, novel techniques have been proposed to improve the efficiency of this task as follows. First, by populating the BKSPORT knowledge base with aliases and synonyms, the nickname of famous sports character such as CR7, El Pulga appearing in articles could be detected. Second, entity recognition at a more detailed conceptual level is implemented to avoid assigning an entity to a more general concept in ontology such as Person or Player instead of the more specific one, such as Forward or Midfielder. Proper rules are defined to recognize the appearance of the concept’s label just before a named entity in the text. The third improvement is the recognition of shortened name entity. In sports news articles, a shortened name of an entity can appear, especially when the full name was used previously. Finally, BKSPORT can disambiguate entities sharing labels but belonging to different types by matching the token standing right after a named entity with concepts in the ontology.

Encouraging results in named entity recognition create the premise for our automatic semantic annotation method. Our approach is to build a set of recognition patterns for important semantics in a form that combine named entity with the predicates defined in the BKSPORT Ontology. Different from other studies that only try to annotate the named entities in documents with its type, the proposed method is able to generate automatically annotation about a number of essential semantics that readers are most interested in sports articles. They are semantic annotations about simple events in the form of triple, important entities, indirect speech, and football transfer events. The detail of this works is described in our previous papers [11][12].

6. TRANSFORMATION FROM NATURAL LANGUAGE QUESTION INTO SPARQL QUERY

To exploit the above semantic annotations, the semantic search needs to be applied. SPARQL, the W3C’s standard query language for RDF, is evidently selected to express input questions. The BKSPORT Semantic Search system comprises two components: (1)
the first component takes a charge of translating question from natural language form into SPARQL query, (2) the second component carries out searching on semantic data using obtained SPARQL queries to receive related news associated with the answers to questions. Because of the availability of many semantic search engines, the implementation of the second component is out of scope of this work. This section presents our efforts for implementing the first component. We aim at developing a user-friendly search system that supports natural language input and does not requires users to have any knowledge about SPARQL.

Because of the complex structure of natural language questions, it is significant to classify these questions before interpreting and translating them to semantic queries. In this work, questions are divided into the following type: predicative questions, yes/no questions, definition questions, association questions, quantity question, multi-subject and multi-object questions, opinion questions, comparative and superlative questions. Moreover, the support for imperative sentences is also envisaged.

![Fig. 3 Question transformation process](image)

Figure 3 depicts the main steps of the process that the proposed method performs to transform a user’s natural language question into a SPARQL query. In the Question Pre-processing step, the user question about news articles expressed using natural language is normalized to support the next steps to function effectively. After the stage of Syntax Analysis where the input question is interpreted syntactically and analyzed grammar elements and its relations, the Semantic Representation of Question creates its corresponding semantic model. From that, the intermediate SPARQL query is generated. The next step annotates variables in this generated query with the help of the ontology and the knowledge base of the system and all variables in the semantic model are identified. The complete SPARQL query is generated simply by filling up fulfilling in the intermediate SPARQL query with corresponding URIs.

### 6.1. QUESTION PRE-PROCESSING AND SYNTAX ANALYSIS

Abnormal tokens normalization and question temporal property determination are the main tasks of the Pre-processing step. The first one detects abbreviations in user questions using a lookup table. In the second task, temporal labels are made statistics and classified into specific types including one-day period, one-week period, one-month period, and one-year period. The system calculates a specific temporal value for each temporal label based on the time readers input a question for further use in the next steps. The transformation from imperative sentences into equivalent standard question form, which is either the wh-question or the yes/no question, is also carried out in this step.

Syntax analysis is an important step that largely affects the quality of the method, as it identifies the question type, grammar elements and the relationship between them. We
conduct NLP tasks such as POSTagged, Phrase Structure Tree and Typed Dependencies analyses using Stanford's CoreNLP toolkit [4] and use the results in many tasks of the following steps. Among them, the Typed Dependencies are most important because they represent the grammatical relations between words in a sentence. Each Typed Dependency is a triplet: name of the relation, governor and dependent. In the proposed method, only certain Typed Dependencies presenting constraints between the elements in question, such as subject-verb, verb-object, passive verb-agent, word-accompanied preposition, noun-adjective modifier are interested. On the basis of this, constraints by triple patterns are formulated in the SPARQL query.

6.2. SEMANTIC REPRESENTATION MODEL OF QUESTION

As shown in Table I, the semantic representation model is defined to cover two basic question forms: wh-question and yes/no question and to play the role of intermediate representation in the query translation process.

Table I. The semantic representation model of question.

<table>
<thead>
<tr>
<th>Variable list:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query variable (quantity query, object query).</td>
</tr>
<tr>
<td>Normal variables</td>
</tr>
<tr>
<td>Constraints of variables:</td>
</tr>
<tr>
<td>Label constraint of variables.</td>
</tr>
<tr>
<td>Constraints of dependent relation between variables.</td>
</tr>
<tr>
<td>Quantity constraints</td>
</tr>
<tr>
<td>Constraints of dependent relations:</td>
</tr>
<tr>
<td>AND/OR constraint.</td>
</tr>
<tr>
<td>Temporal constraint.</td>
</tr>
</tbody>
</table>

Each variable represents a token in the question and has the name in the format “symbol string + ID” (for example: ?x1, ?x2, etc.). There are two types of variables, in which query variables are hidden variables containing information to be queried for the question and ordinary variables are the rest. The difference between a query variable and an ordinary variable is the appearance of “?” symbol at the beginning of their name. While for wh-questions, at least one query variable is compulsory in the variable list, no one is required for yes/no question. We divide query variable into two types. The quantity query variable is represented in variables list as COUNT(?variable_name) for wh-question of “how many” or "how much". The object query variable, represented as ?variable_name, is used for wh-question of “who/what/which/where”. Two main types of constraints are defined in the model are constraints of variable and constraints of dependent relations.

The transformation process from the grammar structure into the semantic representation consists of four steps: identification of query variable, identification of ordinary variables and constraint of dependent relationship between variables and identification of quantity and temporal constraints.

6.2.1. QUERY VARIABLES IDENTIFICATION: query variables are only identified for wh-questions, of which the subject is identified as corresponding to query words. For “who, what and where” questions, the subject is determined as the query word, while for “how many/how much and which” questions, it is identified as the noun following a query word by which the query variable is identified. The type of query variable is identified as corresponding to the query word type.

6.2.2 IDENTIFYING ORDINARY VARIABLES AND CONSTRAINT OF DEPENDENT RELATIONSHIP BETWEEN VARIABLES: From the Typed Dependencies analysis [4] result of the input question, related words and relations among
them (name of Typed Dependency) are inferred mutually. These words are then represented by query and ordinary variables.

The constraints of the question are defined by the triple set, a crucial part of a SPARQL query. However, each dependent relationship between variables, which are identified from Typed Dependency, is only in the double form. These relationships belong to one of two types, subject-predicate or predicate-object. Thus, we proposed to combine Typed Dependencies into triple patterns to generate these constraints. Two different Typed Dependencies which share predicate are combined to create the triple relationship in the form (subject, predicate, and object).

Figure 4 demonstrates the combination of these Two Typed dependencies into a triple pattern in a question with simple semantics such as S-V-O, for example, “Did Barcelona defeat Chelsea?”.

![Fig. 4 Two typed dependencies become a triple pattern.](image)

### 6.2.3. QUANTITY CONSTRAINT IDENTIFICATION:

In the proposed method, two types of quantity constraints are considered. Constraints belonging to the first type (1) compare the number of certain objects with a specific value, i.e., “Who scored more than 3 goals”. The second type (2) represents superlative quantity constraints of a certain object, i.e., “Who scored the most goals,” or “Which team conceded the least goals?”. The algorithm for identifying quantity constraint of type (1) is shown in Figure 5.

![Fig. 5 Identifying quantity constraint of type (1).](image)

For type (2), the detection of “the most/least” + noun structure can be based on two Typed Dependencies, det(?object, “the”) and amod(?object, ?dep). If the value of ?dep in the Typed Dependency admod is “most”, it means the ?object object has the most occurrences and the phase generates themost(?object) constraint in the semantic model. On the contrary, if this value is “least”, the phase generates theleast(?object) constraint.
6.2.4. IDENTIFICATION OF TEMPORAL CONSTRAINT: As the information about time is important in sports events, the system is expected to be able to respond to questions with temporal constraints, which are classified into two types: constraint tied to a time and constraint tied to duration. For example:

- Type (1): “today,” “yesterday,” “last Sunday,” “in 01/01/2018,” and so on.
- Type (2): “last week,” “last month,” “this season,” “this year,” and so on.

To represent temporal semantics for questions, we define one “Interval,” including two fields, in the semantic model: Interval (BEGIN, END). The interval type shows the constraint that the time of events must be in the form of from BEGIN to END. If the input question mentions temporal context, then the syntax analyzer generates Typed Dependency prep_in (object, time_label), where time_label is the temporal label to show the temporal context of question. If time_label is a certain day, BEGIN and END attain the same value. If time_label is a certain week, month, year or season, we depend on the time user making queries to compute BEGIN and END value.

6.3. GENERATION OF INTERMEDIATE SPARQL QUERY

An intermediate SPARQL query constitutes frames containing variables, which consist of two main elements, namely, question clause and condition clause. For special questions with quantity constraint or temporal constraint, there are additional constraint clauses. Figure 6 below illustrates the process to generate intermediate SPARQL queries.

The question clause consists of two types, i.e., SELECT clause or ASK clause, corresponding to two basic question types: wh-question and yes/no question. The ASK query is identified when there is no query variable in the semantic model. Depending on the types of query variables which are quantity query variable (COUNT (?x)) or object query variable (?x), the proper question clause can be generated.

6.3.1. CONDITION (WHERE) CLAUSE GENERATION: The first step of this task is the generation of triples and the representation of their relationship based on AND/OR constraint. The WHERE clause contains triple patterns which are RDF triples in the form of {<?subject> <?predicate> <?object>}. These triples are built based on triple relationships (?subject, ?predicate, ?object) in the semantic model. A triple relationship generates a triple in the WHERE clause.

For instance, if the semantic model contains a triple (?x, ?y, ?z), a triple in the form of (<?x> <?y> <?z>) is generated. The combination of these triples in the WHERE clause (simultaneously or not simultaneously) depends on the AND/OR constraint in the semantic model. If the semantic model contains two triple relationships such as Triple1 (?x1, ?y1, ?z1) and Triple2 (?x2, ?y2, ?z2), and there is AND (Triple1, Triple2) constraint between them, the two triples are transformed into 2 RDF ones going together in the WHERE clause as follows:

\{<?x1> <?y1> <?z1> .<?x2> <?y2> <?z2> .\}.
On the other hand, if the OR (Triple1, Triple2) constraint exists, these two relationships are represented in the union form as follows:

\{
  \{<?x1> <?y1> <?z1>\} UNION \{<?x2> <?y2> <?z2>\}.\}

The second step is the generation of quantity constraints which are represented by GROUP BY clause with supplementary clauses, i.e., HAVING and ORDER.

The HAVING clause is used to represent constraints comparing the amount of a certain object with a specific value. If there is the moreOREqual (?object, 3) constraint in the semantic model, we represent it in the SPARQL query as follows:

GROUP BY ?object
HAVING (COUNT(?object) >= 3).

The ORDER clause represents the quantity constraint of a certain object, which is the most or the least. If there is the themost(?object) constraint in the semantic model, we represent it in the SPARQL query as follows:

GROUP BY ?object
ORDER BY DESC(COUNT(?object)) OFFSET 0 LIMIT 1.

“DESC” keyword shows query results are sorted in the descending order by the COUNT(?object) value. On the contrary, “ASC” means query results are sorted in the ascending order, as it is used in the least(?object) constraint. “OFFSET 0 LIMIT 1” constraint in this example means only the ?object result with the most COUNT(?object) value is returned.

The third step is the representation of Temporal Constraint in SPARQL query. The temporal constraint considered in this work is a dependent relationship constraint between variables. To express temporal constraint for triple patterns, the NamedGraph model is used to group triple relationships in one ?graph graph as follows:

?graph

\{ //RDF triples

\}

Thereafter, temporal constraint for RDF triples is defined through the definition of temporal constraint for the ?graph graph as follows:

?g <http://bk.sport.owl#hasTime> ?t.
?t time:inXSDDate <instantDate>.
FILTER (?instantDate >= “BEGIN”^^<xsd:dateTime> && ?instantDate <= “END”^^<xsd:dateTime> ).

Where, BEGIN and END are two values identified from the Interval(BEGIN, END) constraint in the semantic model of the question.

6.4 ENTITY, CLASS AND PREDICATE RECOGNITION

To get a complete SPARQL query, it is necessary to identify the value and the type of ?x, ?y, ?z variables in the intermediate SPARQL query. This step lookups each variable in the knowledge base and ontology to detect corresponding entity, class or property. The named entity recognition is carried out exactly the same as mentioned in Section 5. If a variable is recognized as an entity in the knowledge base, it is replaced with an URI of the entity. In case a variable label is a proper noun but not recognized, the FILTER statement is inserted into an intermediate SPARQL query to a bind label value to the variable.

To verify if a variable corresponds to a class or a predicate, we first create two lookup tables including two fields; the first is the URIs of all classes/predicates in ontology and the second is their labels. Second, WordNet is used to find synonyms of labels of each of the above URIs, generating a set of representing words for each URI. Finally, the system
examines which set of representing words the labels of each variable belong to. Then, the system identifies corresponding URIs to variables and adds a triple pattern with the syntax of `<variable_name> <rdf:type> <class_URI>` into a SPARQL query if the variable is identified as a class.

7. RESULTS AND EVALUATION

The first version of BKSport was implemented and deployed, fulfilling the most important functionalities as expected. What users can see and interact with is News website, the GUI component of the system, which is developed using Ruby on Rails framework. Other components such as Semantic Annotation System, Semantic Search, Recommender were implemented using Java and function as background services and process. To execute generated SPARQL queries, AllegroGraph engine is used. Figure 7 gives certain screenshots of the BKSport news website that illustrate its main features. It is a freely accessible web application allowing a user to enter natural language questions. The home page features the highlight news items on top and the list of most updated news below as other news systems.

![Screenhots of BKSport news website](image)

The system can be configured to show out not only the SPARQL query translated from a search question in form of natural language but also the visualization of semantic annotation, embedded in the article content. Through the visualization feature, users can see the type (ontology class) associated with entities in a news item and even observe a semantic triple connecting two entities simply by hovering the mouse over the text. Additionally, for a question such as “News about Lionel Messi”, beside a list of news
articles that are semantically relevant, the supplementary information about the entity from the knowledge base is also returned by Semantic Search component.

Unlike general domain, biomedical science or music where QALD datasets are available for evaluating question answering tasks, no sample of questions has been introduced. To evaluate the method for transforming input question into SPARQL queries, we built a set of questions, as shown in Table 2, by simulating the nature of QALD questions but it may differ with regard to the related named entities, subjects or predicates and the question classification as mentioned in Section 6. Accordingly, our experiment dataset consists of 47 typical questions.

For each question, an evaluator with knowledge of the SPARQL can decide the generated SPARQL query is correct or not by checking for the accuracy and the completeness of its elements. In our opinion, the evaluation of SPARQL query based on the “absolute correctness” is not enough, because in some case the output lacks some information but the main semantics of the question are still reserved.

We consider a SPARQL query constitutes three main clause types: question clause, WHERE clause, and other constraint clauses. In order to measure the precision of query, we first measured the precision of each type of clause based on unit elements. A “correct unit element,” is defined to be a variable satisfying one of the following conditions:

- Recognized, corresponding to one URI
- Type is identified clearly
- Constraints on label value are identified clearly

The evaluation metric for the precision of an output query based not only on the satisfied number of elements, but also the importance of each element. Thus, each type of clause in the query is associated to a weight based on our opinion about its importance.

Considering \( w_i \) as weight of a type of clause \( i \) in the query, \( n_i \) as the number of correct unit elements of a type of clause \( i \), \( N_i \) is the number of unit elements of the type of clause \( i \) needed to identify in the pre-written corresponding SPARQL query for that question. Then, \( (n_i/N_i) \) is the precision of the type of clause \( i \). If there is no specific question variable in “SELECT *” question clause, we assign 0.5 to the precision of question clause. Finally, we proposed the formula to measure the precision of a query \( q \) generated by the system as follows:

\[
\text{Precision}(q) = b \times \sum_{i=1}^{M} \left( a_i \times w_i \times \frac{n_i}{N_i} \right) / \sum_{i=1}^{M} a_i w_i
\]

In which:

- \( b \) gets the value of 0 or 1
  - \( b = 0 \), if identification of question clause (SELECT or ASK) is wrong, or wrong identification of all question variables occurs. Question variables decide which object needs to be queried. If all question variables are identified wrong, precision = 0
  - \( b = 1 \) in the other cases
- \( M \) is the number of clause types in the query written by experts
- \( a_i = 1 \), if the clause exists in the query generated by the system, \( a_i = 0 \) otherwise

Table II illustrates a part of the experiment dataset of questions and the corresponding precision computed using the proposed formula. The overall precision over the full set of questions is measured as the mean of the precision value of all questions, and gets finally the encouraging value of 91.89%.
Table II. A part of the evaluation questions set.

<table>
<thead>
<tr>
<th>ID</th>
<th>Question</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Who is Lionel Messi</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Was Chelsea defeated by Barcelona?</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Was Chelsea defeated by Barcelona last year?</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>Does Lionel Messi play for Barcelona?</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Did Barcelona defeat Chelsea?</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>Did Barcelona defeat Chelsea last year?</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>Did Wayne Rooney dispute with Alex Ferguson yesterday?</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>Which team defeated Chelsea?</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>Which team defeated Chelsea this season?</td>
<td>0.83</td>
</tr>
<tr>
<td>11</td>
<td>Which event relates to Lionel Messi?</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>Which news is published today?</td>
<td>0.67</td>
</tr>
<tr>
<td>13</td>
<td>What happened to Lionel Messi?</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>Which team did Lionel Messi transfer to?</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>What team does Lionel Messi play for in 2010?</td>
<td>0.83</td>
</tr>
<tr>
<td>16</td>
<td>What did Lionel Messi say about Chelsea?</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>News about Chelsea</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>How many games has Lionel Messi play in his career?</td>
<td>0.87</td>
</tr>
<tr>
<td>19</td>
<td>How many clubs defeated Chelsea?</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>Which team won 3 games this years?</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>Who won the most games this year?</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>Who won more than 1 game this year?</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>Which team has at least 22 players?</td>
<td>0.89</td>
</tr>
<tr>
<td>24</td>
<td>What is the result of the match between Chelsea and Barcelona?</td>
<td>1</td>
</tr>
<tr>
<td>25</td>
<td>What is the result of the match between Chelsea and Barcelona yesterday?</td>
<td>1</td>
</tr>
<tr>
<td>26</td>
<td>What happened between Chelsea and Barcelona?</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>Which team defeated Chelsea and Barcelona?</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>Which team defeated Chelsea and Barcelona in 2014?</td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>Did Manchester United and Chelsea defeat Barcelona?</td>
<td>1</td>
</tr>
<tr>
<td>30</td>
<td>Did Manchester United and Chelsea defeat Barcelona in 2014?</td>
<td>1</td>
</tr>
<tr>
<td>31</td>
<td>Was Barcelona defeated by Manchester United and Chelsea?</td>
<td>1</td>
</tr>
<tr>
<td>32</td>
<td>Tell me where FIFA is located in</td>
<td>0.67</td>
</tr>
<tr>
<td>33</td>
<td>Tell me which team Lionel Messi transferred to</td>
<td>1</td>
</tr>
<tr>
<td>34</td>
<td>Give me all soccer clubs in the Premier League</td>
<td>0.6</td>
</tr>
<tr>
<td>35</td>
<td>Wayne Rooney disputed with whom yesterday?</td>
<td>0.64</td>
</tr>
<tr>
<td>36</td>
<td>Which player will leave Chelsea?</td>
<td>0.89</td>
</tr>
</tbody>
</table>

6. CONCLUSIONS AND FUTURE WORK

We have presented a news aggregation system that applies Semantic Web technologies to provide advanced functionalities including automatic semantic annotation creation, semantic search with natural language questions. The system was built around an ontology and knowledge base, developed particularly for the domain of sports news. Besides providing semantic primitives for representing news items, they are the key to boost up the effectiveness of many components. Methods are proposed to create semantic annotations of news articles and to transform a natural language query into a SPARQL
query. As a result, users can ask the website about the news articles in a convenient way, posing question formalize in natural language.

The novelty of our methods lies in the combination of Semantic Web technology and NLP techniques for domain-specific ontology, knowledge base construction and a pattern-based approach. Our experience showed that it is possible to apply semantic technologies to enhance the most important features of a news system like searching, visualization. Based on grammar structure deep analysis of questions, the proposed method can handle complex forms of questions that are not supported in other works such as comparative question, superlative question, questions with multiple subjects and objects, questions with abnormal grammar structure, questions with temporal constraint, and so on.

In the future, the authors focus on improving the BKSport ontology to cover concepts and relationships in the sports domain in a more sufficient manner. Second, machine learning and deep learning techniques are considered to overcome the shortcomings of rule-based methods when they work on data sets that are big on volume and change rapidly.

REFERENCES


