An Ontology-Based Retrieval System Using Semantic Indexing

Soner Kara [#1], Özgür Alan [#2], Orkunt Sabuncu [#3], Samet Akpınar [*4], Nihan K. Çiçekli [*5], Ferda N. Alpaslan [*6]

[#] *Orbim Corp.*
METU Technopolis, Ankara, Turkey
[1] soner.kara@orbim.com.tr
[2] alan@ceng.metu.edu.tr
[3] orkunt@ceng.metu.edu.tr

[*] *Dept. of Computer Engineering*
METU, Ankara, Turkey
[4] samet@ceng.metu.edu.tr
[5] nihan@ceng.metu.edu.tr
[6] alpaslan@ceng.metu.edu.tr

***Abstract*— In this paper, we present an ontology-based information extraction and retrieval system and its application to soccer domain. In general, we deal with three issues in semantic search, namely, usability, scalability and retrieval performance. We propose a keyword-based semantic retrieval approach. The performance of the system is improved considerably using domain-specific information extraction, inference and rules. Scalability is achieved by adapting a semantic indexing approach. We implement the system using the state-of-the-art technologies in Semantic Web and evaluate the performance against traditional systems. Further detailed evaluation is provided to observe the performance gain due to domain-specific information extraction and inference.**

I. Introduction

The huge increase in the amount and complexity of reachable information in the World Wide Web caused an excessive demand for tools and techniques that can handle data semantically. Current practice in information retrieval mostly relies on keyword-based search over full-text data, which is modeled with bag-of-words. However, such a model misses the actual semantic information in text. To deal with this issue, ontologies are proposed [1] for knowledge representation, which are nowadays the backbone of semantic web applications. Both the information extraction and retrieval processes can benefit from such metadata, which gives semantics to plain text.

Having obtained the semantic knowledge and represented them via ontologies, the next step is querying the semantic data, also known as semantic search. There are several query languages designed for semantic querying. Currently, SPARQL is the state-of-the-art query language for Semantic Web. Unfortunately, these formal query languages are not meant to be used by the end-users. Formulating a query using such languages requires the knowledge of the domain ontology as well as the syntax of the language. Therefore, Semantic Web community works on simplifying the process of query formulating for the end-user. Current studies on semantic query interfaces are carried in four categories, namely, keyword-based, form-based, view-based and natural language-based systems as reviewed in [2]. Out of these, keyword-based query interfaces are the most user-friendly ones and people are already used to use such interfaces thanks to Google.

Combining the usability of keyword-based interfaces with the power of semantic technologies is one of the most challenging areas in semantic searching. According to our vision of Semantic Web, all the efforts towards increasing retrieval performance while preserving user-friendliness will eventually come to the point of improving semantic searching with keyword-based interfaces. This is a challenging task as it requires complex queries to be answered with only a few keywords. Furthermore, it should allow the inferred knowledge to be retrieved easily and provide a ranking mechanism to reflect semantics and ontological importance.

In this paper, we present a complete ontology-based framework for extraction and retrieval of semantic information in limited domains. We applied the framework in soccer domain and observed the improvements over classical keyword-based approaches. The system consists of an automated information extraction module, an ontology populator module, an inference module, and a keyword-based semantic query interface. Our main concern, in this study, is achieving a high retrieval performance while preserving user-friendliness. We show that our system achieves very high precision and recall values even for the very complex queries a user can ask in soccer domain. Furthermore, we evaluate and report the effects of information extraction and inference on query performance.

The rest of the paper is organized as follows: A brief discussion about the related work is given in Section II. In Section III, we give the details of the components of the system, namely IE, ontology population, inference and searching. In Section IV, we report the experiments and their results. Section V concludes the paper with some remarks for future work.

978-1-4244-6523-1/10/$26.00 © 2010 IEEE

II. Related Work

Classical or traditional keyword-based information retrieval approaches are based on the vector space model proposed by Salton et al. [3]. In this model, documents and queries are simply represented as a vector of term weights and retrieval is done according to the cosine similarity between these vectors. [4], [5], [6] and [7] are some of the important studies related to traditional searching. This approach does not require any extraction or annotation phase. Therefore, its easy to implement, however, the precision values are relatively low.

The first step towards semantic retrieval was using WordNet synonym sets (synsets) for word semantics [8], [9]. The main idea was expanding both the indices and queries with the semantics of the words to achieve better recall and precision. If used together with an effective word sense disambiguation (WSD) algorithm, this approach is shown to improve retrieval performance. On the other hand, a poor WSD will cause degradation in performance. Another drawback of this approach is the lack of complex semantics as it is limited with the relations defined in the WordNet.

With the introduction of semantic web technologies, knowledge representation has become more structured and sophisticated, which requires more advanced extraction and retrieval methods to be implemented. The general approach is storing the extracted data in RDF or OWL format, and querying with RDF query languages such as RDQL or SPARQL. Although this approach offers the ultimate precision and recall performance, it is far from useful since it requires a relatively complex query language.

To overcome the difficulties of learning a formal query language, a number of query interface methods are proposed [2]. As we stated earlier, our main focus is keyword-based interfaces. There are several approaches to implement keyword-based querying. To mention a few, SPARK [10] uses a probabilistic query ranking approach for constructing the best query represented by the keywords. Q2Semantic [11] tries to find the best sub-graph expressing the query in the RDF graph. SemSearch [12] uses a template-based approach for query construction. These approaches are not easily scaled to large knowledge-bases as they require traversing RDF graphs or querying the same knowledgebase several times for a single search.

A scalable alternative to query construction from keywords is semantic indexing. In this approach, semantic data in RDF knowledge-bases are indexed in a structured way and made directly available to use with keyword queries. [13], [14] and [15] adapt a similar approach. They index all extracted RDF triples together with the corresponding free text. Since they use very basic extraction methods, such a naive indexing seems feasible. However, complex semantics cannot be captured from the indices containing only subject-predicate-object triples as index terms. If a retrieval system should answer complex queries involving extracted and inferred knowledge, the index must be designed and enriched accordingly.

Our literature survey revealed that current studies on the keyword-based semantic searching are not mature enough: Either they are not scalable to large knowledgebases or they cannot capture all the semantics in the queries. Our aim is to fill this gap by implementing a keyword-based semantic retrieval system using the semantic indexing approach. In other words, we try to implement a system that performs at least as good as traditional approaches and improves the performance and usability of semantic querying. We tested our system in soccer domain to see the effectiveness of semantic searching over traditional approaches and observed a remarkable increase in precision and recall. Moreover we noted that our system can answer complex semantic queries, which is not possible with traditional methods. The study presented in this paper can be extended to other domains as well by modifying the current ontology and the information extraction module as described in [16].

III. Our Approach To Semantic Retrieval

Within the scope of this paper we have developed a fully fledged semantic application which a) contains all the aspects of SW from information extraction to information retrieval and b) uses all the cutting-edge technologies such as OWL-DL, inference, rules, RDF repositories and semantic indexing. The overall diagram of the framework can be seen in Fig. 1.

A. Ontology Design

We designed a central soccer ontology, which is utilized by every aspect of the system, especially in information extraction, inference and retrieval phases. Thus, the overall performance of the system is highly dependent on its quality. We followed an iterative development process in the ontology engineering phase. First, we started with a core ontology including basic concepts and a simple hierarchy. Then, we experimented with this ontology and fix the issues in reasoning and searching. These steps were repeated until we end up with a stable ontology containing 79 classes and 95 properties in soccer domain.

B. Information Extraction (IE)

Information extraction is one of the most important parts of ontology-based semantic web applications. It is the process of adding structured information to the knowledgebase by processing unstructured resources. In this phase, we use the data crawled from the Web sites such as UEFA[1] and SporX[2]. What we obtain as the output of web crawler is some basic information specific to a match (teams, players, goals, stadium, etc.) and natural language texts (narrations of that match). The basic information and the narrations are used as input to our information extraction (IE) module. The details of this module are reported in [16] by Tunaoglu et al. Basically, it is a template-based IE approach for specific domains.We can achieve 100% success rate in UEFA narrations thanks to the language used in the UEFA web-site, which is highly structured and error-free. Fig. 2 gives an idea about the

[1]http://www.uefa.com/
[2]http://www.sporx.com/

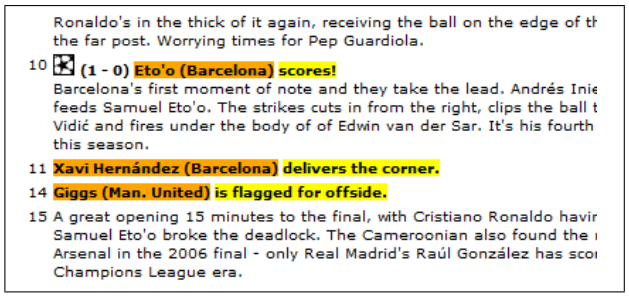

Fig. 2. Example extractions from UEFA narrations

information we can extract from the UEFA web-site. The integration of this module to the system is done in a loosely coupled fashion, so we can use it in semantic applications for any language or domain.

C. Ontology Population

Ontology population is the process of knowledge acquisition by transforming or mapping unstructured, semi-structured and structured data into ontology instances [17]. Our information extractor module [16] already does most of the labor by extracting structured information from unstructured text narrations. For example, from the narration "Keita commits a foul after challenging Belletti" we obtain a foul object, more specifically `FOUL(Keita, Belletti)`. Having the output of the IE module, the ontology population process now becomes creating an OWL individual for each object extracted during IE.

If the IE module cannot extract some attribute of an event, we still create an instance with empty properties. Thus, the recall performance for simple queries will not be affected even IE fails to extract some details of the event. Moreover, if no event is detected in a narration, an instance with the type `UnknownEvent` is created. Unknown events are not discarded because of the reasons described in Section III-E.1. Fig. 3 shows the process of ontology population starting from UEFA narrations ending with OWL instances.

Ontology population is not restricted with the events extracted from the IE module. As mentioned earlier, the crawled information also contains some basic information about the match including players, teams, referees, stadium etc. This information is also added to the ontology by creating an OWL individual for each of them if they do not already exist in the knowledgebase.

D. Inference and Rules

The formal specification of Web Ontology Language, OWL, is highly influenced by Description Logics (DLs)[3]. OWL-DL is designed to be computationally complete and decidable version of OWL, thus it benefits from a wide range of sound, complete and terminating DL reasoners. For our inference module, we use Pellet[4], an open-source DL-reasoner, which supports all the standard inference services such as consistency checking, concept satisfiability, classification and realization.

Consistency checking ensures that there is no contradictory assertion in the ontology. In order to benefit from this feature, we specify some property restrictions during the ontology development. There are two kinds of restrictions in OWL: value constraints and cardinality constraints. We use value constraints, for example, to state that only the goalkeepers (a subset of players) are allowed in the position of goalkeeping and using a cardinality constraint, we can say that only one goalkeeper is allowed in the game. These restrictions not only are useful in consistency checking but also allow new information to be inferred. For example, we could infer the type of an individual if it is the value of a property whose range is restricted to a certain class.

Using classification reasoning we obtain the whole class hierarchy according to class-subclass definitions in the ontology. Inferring new knowledge through classification is a domain independent process and its contribution to the knowledgebase is trivial. In order to infer more interesting information, we use Jena[5] rules.

To illustrate the power of Jena Rules, we give the example of inferring an Assist event. Using the Jena rule shown in Fig. 4, we are able to add a new Assist instance to our knowledgebase. The rule simply looks for two events, namely a Goal and a Pass, that happened in the same match in the same minute and the receiver of the pass is the same person with the scorer. If this is the case, then an Assist instance is created and added to the knowledgebase.

```
noValue(?pass rdf:type pre:Assist)
(?pass rdf:type pre:Pass)
(?pass pre:passingPlayer ?passer)
(?pass pre:passReceiver ?receiver)
(?pass pre:inMatch ?match)
(?pass pre:inMinute ?minute)
(?goal pre:inMatch ?match)
(?goal pre:inMinute ?minute)
(?goal pre:scorerPlayer ?receiver)
makeTemp(?tmp)

-> (?tmp rdf:type pre:Assist)
   (?tmp pre:inMatch ?match)
   (?tmp pre:inMinute ?minute)
   (?tmp pre:passingPlayer ?passer)
   (?tmp pre:passReceiver ?receiver)
```

Fig. 4. An Example for Jena Rules (Assist rule)

E. Semantic Indexing and Retrieval

For the retrieval part, we adapt a semantic indexing approach based on Lucene[6] indices. The idea is extending traditional full-text index with the extracted and inferred knowledge and modifying the ranking so that documents containing ontological information gets higher rates. The details of the index structure and ranking are given in Section III-E.1 and III-E.2 respectively.

[3]http://www.w3.org/TR/owl-guide/
[4]http://clarkparsia.com/pellet
[5]http://jena.sourceforge.net/
[6]http://lucene.apache.org/

TABLE I
INDEX STRUCTURE (SIMPLIFIED FOR BETTER UNDERSTANDING)

Field	Value
docNo	7
event	Foul
match	Chelsea_Barcelona_06_05_2009_20_45
team1	Chelsea
team2	Barcelona
date	2009-05-06
minute	43
subjectPlayer	Michael Ballack
subjectTeam	–
objectPlayer	Sergio Busquets
objectTeam	–
narration	Ballack gives away a free-kick following a challenge on Busquets

1) Index Structure: The structure of semantic index has utmost importance in the retrieval performance. We constructed a Lucene index such that each entry represents a soccer event. As we have mentioned in the previous sections, each event has its own properties associated with it, such as subjects and objects. That information is also included with each event. We also include full-text narrations associated with events to the index. This is especially important if the event type is unknown (an event which is not recognized by the information extractor). Adding full-text narrations to the index tolerates the incomplete event information, thus ensures at least the recall values of traditional full-text search. The index structure can be seen with an example entry in Table I.

2) Searching and Ranking: In traditional keyword search, indexed documents usually contain nothing but raw text associated with that document. Lucene can easily handle such indices and its default ranking gives usually good results. However, complex indices should be handled carefully. In order to take the advantages of our ontology-aided index structure, we slightly modified default querying and ranking mechanism of Lucene. First of all, we boosted the ranking of fields containing extracted and inferred information to stress the importance of them. Secondly, these fields are re-ranked according to their importance. For example, the "event" field is given the highest ranking. This approach prevents misleadings stemming from ambiguous words in full-text. For example, lets say a narration contains "Ronaldo misses a goal": Searching for a "goal" in a traditional search may return this document in the first place, which is a false positive. However, in ontology-aided index, the events whose type is `Goal` will have higher ranks. Since the type of the event above is a `Miss`, it will have a lower rank.

IV. EVALUATION

In order to evaluate the retrieval performance of our system, we have crawled 10 UEFA matches, containing a total of 1182 narrations. Out of these narrations, our IE module was able to extract 902 events. Using these data, we constructed 4 Lucene indices for detailed comparisons. First, we built a traditional full-text index, `TRAD`, using only the narrations of the UEFA matches. This index is used as the baseline for the performance of other methods. Then, we built 2 indices for ontology aided semantic search, namely `BASIC_EXT` and `FULL_EXT`, where the former contains only the basic information available in the UEFA crawl and the latter contains the extracted information in addition to the basic information. Finally, we built an index, `FULL_INF`, which is the expanded version of `FULL_EXT` with the inferred knowledge. All of the indices are evaluated with the queries shown in Table II.

The results can be seen in Table III. First of all, consider the first three queries. There is a considerable difference between `TRAD` and the other methods. The reason is that UEFA narrations use the phrase "P scores!" when the player P scores a goal. Since they omit the word "goal" in narrations, traditional index is not able to retrieve all the goals with the keyword query: "goal". However, the information extraction module can successfully recognize the goal and we can index it as a document with its eventType field filled as "goal". Thus, the improved index can answer both the queries "goal" and "scores" successfully. That is the reason why `BASIC_EXT` and the other indices have very high precision rates.

The improvement provided by the information extraction module can be seen clearly by looking at the difference between `BASIC_EXT` and `FULL_EXT` in 9th and 10th queries. The difference stems from the extracted events such as shoots and goalkeeper saves.

Improvements stemming from the inference can be observed by looking at the queries 4, 7 and 10. In these queries, `FULL_INF` index performs much better than other indices, because it contains additional information due to ontological inference and classification. For example, the 4th query exploits the inferred knowledge about the fact that red cards and yellow cards are also known as punishments. Similarly, the 10th query benefits from the inferred defence players through classification. Finally, the 7th query uses the knowledge obtained from the property hierarchies defined in the ontology. This means, the system can recognize the properties such as `actorOfMissedGoal`, `actorOfOffside`, and `actorOfRedCard` as `actorOfNegativeMove`. Moreover, in the 6th query, we can see the effect of Jena rules. Here, according to one of the rules we defined, we can infer the implicit knowledge of which goal is scored to which goalkeeper, even if that knowledge does not exist explicitly.

However, some queries can slightly suffer from the pollution caused by the information added to the index during the inference. This is mainly due to the fact that some of the fields of `FULL_INF` become very crowded by adding the inferred knowledge, thus slightly deteriorate the rankings. Query-8 illustrates this problem. It is a simple query with a single player name (ronaldo). However, the subjectPlayer field of `FULL_INF` contains some detailed player information in addition to his name. Therefore, the name of the player becomes less significant and the corresponding document is poorly ranked. This problem can be solved by extending the index structure with additional fields rather than accumulating all the information in a single field.

TABLE II
EVALUATION QUERIES

Q-1	Find all goals **(query: goal)**
Q-2	Find all goals scored by Barcelona **(query: barcelona goal)**
Q-3	Find all goals scored by Messi at Barcelona **(query: messi barcelona goal)**
Q-4	Find all punishments **(query: punishment)**
Q-5	Find all yellow cards received by Alex **(query: alex yellow card)**
Q-6	Find all goals scored to Casillas **(query: goal scored to casillas)**
Q-7	Find all negative moves of Henry **(query: henry negative moves)**
Q-8	Find all events involving Ronaldo **(query: ronaldo)**
Q-9	Find all saves done by the goalkeeper of Barcelona **(query: save goalkeeper barcelona)**
Q-10	Find all shoots delivered by defence players **(query: shoot defence players)**

TABLE III
EVALUATION RESULTS (MEAN AVERAGE PRECISION)

	TRAD		**BASIC_EXT**		**FULL_EXT**		**FULL_INF**	
Q-1	0.5/35	1.4%	35/35	100%	35/35	100%	35/35	100%
Q-2	0.4/7	5.7%	5.3/7	75.7%	5.3/35	75.7%	5.3/35	75.7%
Q-3	0.7/3	23.3%	3/3	100%	3/3	100%	3/3	100%
Q-4	0/43	0%	0/43	0%	0/43	0%	43/43	100%
Q-5	1.1/2	55%	2/2	100%	2/2	100%	2/2	100%
Q-6	0.1/9	1.1%	5.7/9	63.3%	5.6/9	62.2%	9/9	100%
Q-7	2.2/7	31.4%	1.9/7	27.1%	2.3/7	32.8%	6.3/7	90.0%
Q-8	7.9/11	71.8%	8.6/11	78.1%	8.5/11	77.2%	7.4/11	67.2%
Q-9	5.1/8	63.7%	4.5/8	56.2%	6.3/8	78.7%	7.5/8	93.7%
Q-10	0/83	0%	0/83	0%	21.9/83	26.4%	81.4/83	98.1%

V. CONCLUSIONS

We have presented a novel semantic retrieval framework and its application to soccer domain, which includes all the aspects of Semantic Web, namely, ontology development, information extraction, ontology population, inference, semantic rules, semantic indexing and retrieval. During the evaluation of the system, we observed that domain-specific information extraction greatly boosts the precision and recall values. Moreover, inference and rules further improve the performance and allow complex domain-specific queries to be handled successfully. Having observed the success in soccer domain, we presume that similar performance can be achieved in other domains as well by extending the ontology and IE module to adapt to the new domains. Indexing and retrieval with keyword interface can be easily adapted to any domain. We also plan to further improve semantic retrieval by adding RDF triples to inferred index without deteriorating the ranking and solve the issues mentioned in Section IV.

ACKNOWLEDGMENT

This work is partially supported by The Scientific and Technical Council of Turkey Grant TUBITAK EEEAG-107E234 and by TUBITAK TEYDEB-3080231.

REFERENCES

[1] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," *Int. J. Hum.-Comput. Stud.*, vol. 43, no. 5-6, pp. 907–928, 1995.

[2] V. Uren, Y. Lei, V. Lopez, H. Liu, E. Motta, and M. Giordanino, "The usability of semantic search tools: A review," *Knowl. Eng. Rev.*, vol. 22, no. 4, pp. 361–377, 2007.

[3] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[4] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, pp. 11–21, 1972.

[5] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[6] G. Salton, E. A. Fox, and H. Wu, "Extended boolean information retrieval," *Commun. ACM*, vol. 26, no. 11, pp. 1022–1036, 1983.

[7] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," in *Information Processing and Management*, 1988, pp. 513–523.

[8] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarrin, "Indexing with wordnet synsets can improve text retrieval," 1998, pp. 38–44.

[9] R. Mihalcea and D. Moldovan, "Semantic indexing using wordnet senses," in *Proceedings of the ACL-2000 workshop on Recent advances in natural language processing and information retrieval*. Morristown, NJ, USA: Association for Computational Linguistics, 2000, pp. 35–45.

[10] Q. Zhou, C. Wang, M. Xiong, H. Wang, and Y. Yu, "Spark: Adapting keyword query to semantic search," in *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC2007), Busan, South Korea*, ser. LNCS, vol. 4825. Berlin, Heidelberg: Springer Verlag, November 2007, pp. 687–700.

[11] H. Wang, K. Zhang, Q. Liu, T. Tran, and Y. Yu, "Q2semantic: A lightweight keyword interface to semantic search," in *ESWC*, 2008, pp. 584–598.

[12] Y. Lei, V. S. Uren, and E. Motta, "Semsearch: A search engine for the semantic web," in *EKAW*, 2006, pp. 238–245.

[13] U. Shah, T. Finin, A. Joshi, R. S. Cost, and J. Matfield, "Information retrieval on the semantic web," in *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*. New York, NY, USA: ACM, 2002, pp. 461–468.

[14] J. Davies and R. Weeks, "Quizrdf: Search technology for the semantic web," *Hawaii International Conference on System Sciences*, vol. 4, pp. 40 112+, 2004. [Online]. Available: http://dx.doi.org/10.1109/HICSS.2004.1265293

[15] I. Celino, E. D. Valle, D. Cerizza, and A. Turati, "Squiggle: An experience in model-driven development of real-world semantic search engines." in *ICWE*, ser. Lecture Notes in Computer Science, L. Baresi, P. Fraternali, and G.-J. Houben, Eds., vol. 4607. Springer, 2007, pp. 485–490. [Online]. Available: http://dblp.uni-trier.de/db/conf/icwe/icwe2007.html#CelinoVCT07

[16] D. Tunaoglu, O. Alan, O. Sabuncu, S. Akpinar, N. K. Cicekli, and F. N. Alpaslan, "Event extraction from turkish football web-casting texts using hand-crafted templates," in *In Proc. of Third IEEE Inter. Conf. on Semantic Computing (ICSC) (in press)*, 2009.

[17] (2008) The semantic web wiki. [Online]. Available: http://semanticweb.org/wiki/Category:Topic_ontology_population

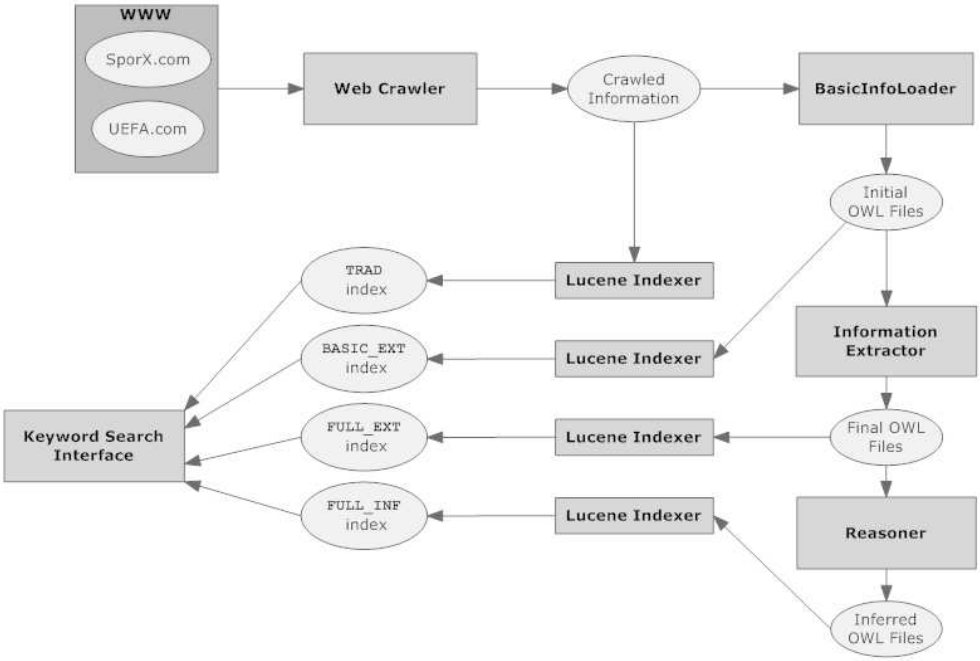

Fig. 1. Overall System Diagram

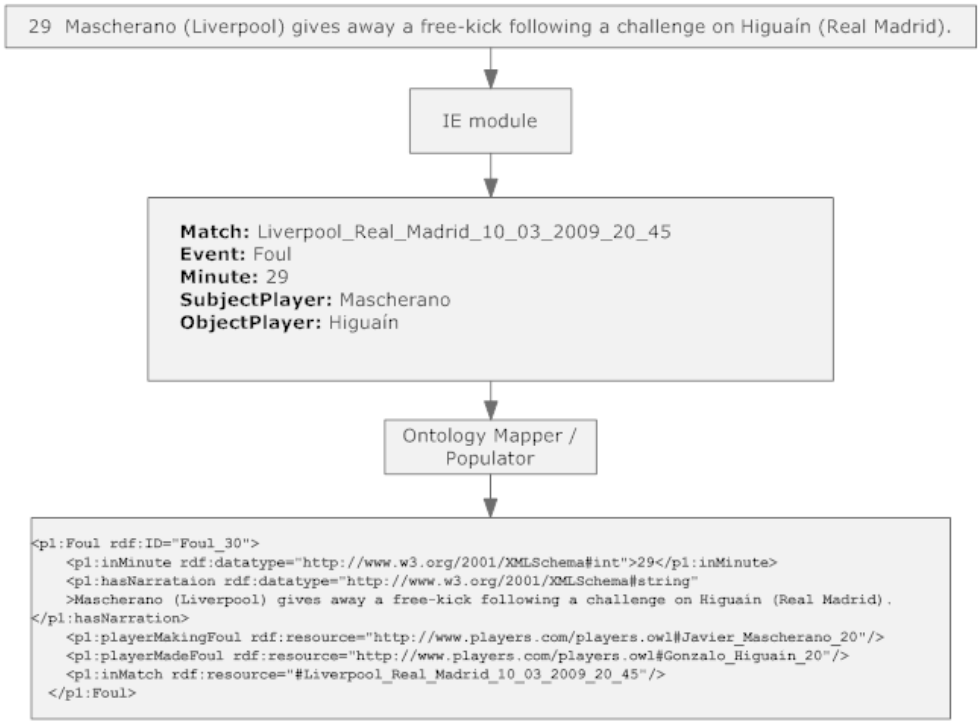

Fig. 3. Information Extraction and Ontology Population