Towards Semantic Web Information Extraction

Borislav Popov, Atanas Kiryakov, Dimitar Manov, Angel Kirilov, Damyan Ognyanoff, Miroslav Goranov

Ontotext Lab, Sirma AI EOOD, 135 Tsarigradsko Shose, Sofia 1784, Bulgaria
{naso, borislav, damyan, mitac, angel, miro}@sirma.bg

Abstract. The approach towards Semantic Web Information Extraction (IE) presented here is implemented in KIM – a platform for semantic indexing, annotation, and retrieval. It combines IE based on the mature text engineering platform (GATE[1]) with Semantic Web-compliant knowledge representation and management. The cornerstone is automatic generation of named-entity (NE) annotations with class and instance references to a semantic repository.

Simplistic upper-level ontology, providing detailed coverage of the most popular entity types (Person, Organization, Location, etc.; more than 250 classes) is designed and used. A knowledge base (KB) with de-facto exhaustive coverage of real-world entities of general importance is maintained, used, and constantly enriched. Extensions of the ontology and KB take care of handling all the lexical resources used for IE, most notable, instead of gazetteer lists, aliases of specific entities are kept together with them in the KB.

A Semantic Gazetteer uses the KB to generate lookup annotations. Ontology-aware pattern-matching grammars allow precise class information to be handled via rules at the optimal level of generality. The grammars are used to recognize NE, with class and instance information referring to the KIM ontology and KB. Recognition of identity relations between the entities is used to unify their references to the KB. Based on the recognized NE, template relation construction is performed via grammar rules. As a result of the latter, the KB is being enriched with the recognized relations between entities. At the final phase of the IE process, previously unknown aliases and entities are being added to the KB with their specific types.

1 Introduction

The acquisition of masses of metadata for the web content would allow various Semantic Web applications to emerge and gain wide acceptance. Such applications would provide and use new access methods based on the associated metadata. The manual semantic authoring, although accurate and sometimes unavoidable, simply does not match the scale as well as the authoring and usage practices typical for the web content. The approach for automatic extraction of metadata is promising scalable, cheap, author-independent and (optionally) user-specific enrichment of the web

[1] General Architecture for Text Engineering (GATE), http://gate.ac.uk, leading NLP and IE platform developed at the University of Sheffield.

content. However, at present there is no technology available to provide automatic semantic annotation in conceptually clear, intuitive, scalable, and accurate enough fashion. Even more, there is no clear vision regarding the approach and model for generation and representation of such annotations.

This paper presents first a model for semantic content enrichment, which we name semantic annotation (section 2.) This model is implemented in a system called KIM and presented in the third section. Most attention is paid to the information extraction (IE[2]) approach used in KIM for automatic semantic annotation; discussed in section 4 with its processing components, KB resources, and resulting linking of NE references to the ontology and KB. Next, evaluation of the performance is presented in the fifth section followed by short overview of related work in section 6. The last section provides a conclusion and discussion on future work.

2 Semantic Annotation

The semantic annotation offered here is a specific metadata generation and usage schema targeted to enable new information access methods and extend existing ones. It is based on the hypothesis that the named entities[3] mentioned in the documents constitute important part of their semantics. Semantic annotation is also the task for/process of generating such metadata.

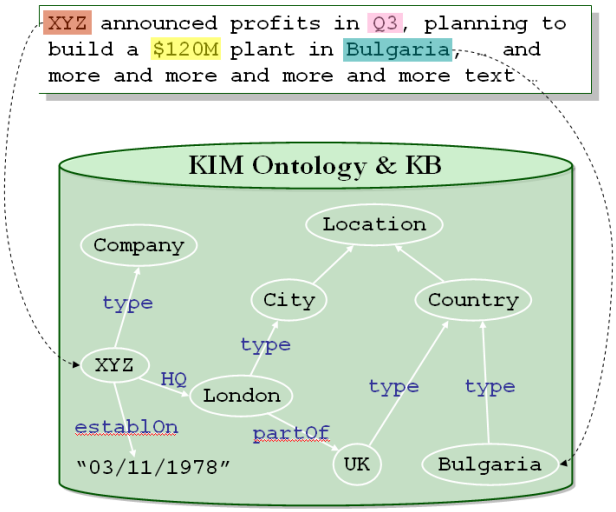


Fig. 1. Semantic Annotation

[2] Information Extraction is a relatively new discipline in the Natural Language Processing (NLP), which conducts partial analysis of text in order to extract specific information.

[3] Named Entities (NE) are people, organizations, locations, and others that are referred by name. The wide interpretation of the term includes any tokens referring something specific in the world: numbers, addresses, amounts of money, dates, etc.

In a nutshell, we consider Semantic Annotation the idea of assigning to the entities in the text links to their semantic descriptions (as presented on Fig. 1). The idea of this sort of metadata is to provide both class and instance information about the entities referred in the documents. It is a question of terminology whether these annotations should be called "semantic," "entity" or some other way. To the best of our knowledge there is no well established term for this task; neither there is a well established meaning for the term "semantic annotation"[4]. What is more important, the automatic semantic annotations enable many new applications: highlighting, indexing and retrieval, categorization, generation of more advanced metadata, smooth traversal between unstructured text and available relevant knowledge. Semantic annotation is applicable for any sort of text – web pages, regular (non-web) documents, text fields in databases, etc. Further, knowledge acquisition can be performed based on extraction of more complex dependencies – analysis of relationships between entities, event and situation descriptions, etc. We believe that, defined this way, semantic annotation is clearly specified, easy to understand, and can serve as a basis for number of useful applications (some of those demonstrated in KIM).

The automatic semantic annotation can be seen as a classical named-entity recognition (NER) and annotation process. The traditional flat NE type sets consist of several general types (such as **`Organization, Person, Date, Location, Percent, Money`**). Although these represent the most important domain-independent NE types, still the entities with same type are dividable in more specific classes from the average educated human (e.g. public companies, sport teams, and syndicates are all organizations). The semantic annotation is specific for providing more precise type information, because the NE type is specified by reference to an ontology. Further, and more important, the semantic annotation requires identification of the entity. While in a classical NER task, guessing the type is everything to be achieved, a semantic annotation needs to recognize the entity (either out of a set of known ones either as unknown one) and refer to it. There is some similarity with the understanding of "content extraction" as used in the context of the ACE project[5].

2.1 Semantic Annotation Model and Representation

Here we discuss the structure and the representation of the semantic annotations, including the necessary knowledge and metadata. There are number of basic prerequisite for representation of semantic annotations:

- Ontology (or at least taxonomy) bearing the classes of entities. It should be possible to refer to the classes in the ontology;
- Unique entity identifiers which allow, those to be identified and linked to their semantic descriptions;
- Knowledge base with entity descriptions.

[4] The term is previously used in [23] in a bit more general sense compared to what we propose, but it didn't get wide acceptance.

[5] See www.itl.nist.gov/iad/894.01/tests/ace/

The next question considers an important choice for the representation of the annotations – "to embed or not to embed?" There are number of arguments providing evidence that the semantic annotations have to be decoupled from the content they refer to. One key reason is to allow dynamic user-specific semantic annotations – the embedded annotations become part of the content and may not change corresponding to the interest of the user or the context of usage. Further, embedded complex annotations (such as those necessary for the Semantic Web) would have negative impact on the volume of the content and can complicate its maintenance – imagine that page with three layers of overlapping semantic annotations need to be updated preserving them consistent. Those and number of other issues defending the externally encoded annotation can be found in [20] which also provides an interesting parallel to the open hypermedia systems.

Once decided that the semantic annotations have to be kept separate from the content, the next question is whether or not (and how much) to couple the annotations with the ontology and the knowledge base? It is the case that such integration seems profitable – it would be easier to keep in synch the annotations with the class and entity descriptions. However, there are at least two important problems:

- Both the cardinality and the complexity of the annotations differ from those of the entity descriptions – the annotations are simpler, but their count is much bigger than this of the entity descriptions. Even considering middle-sized document corpora the annotations can reach tens of millions. Suppose 10M annotations are stored in an RDF(S) store together with 1M entity descriptions. Suppose also that each annotation and each entity description are represented with 10 statements. There is a big difference regarding the inference approaches and hardware capable in efficient reasoning and access to 10M-statement repository and with 110M-statement repository.
- It would be nice if the world knowledge (ontology and instance data) and the document-related metadata are kept independent. This would mean that for one and the same document different extraction, processing, or authoring methods will be able to deliver alternative metadata referring to one and the same knowledge store.
- Most important, it should be possible the ownership and the responsibility for the metadata and the knowledge to be distributed. This way, different parties can develop and maintain separately the content, the metadata, and the knowledge.

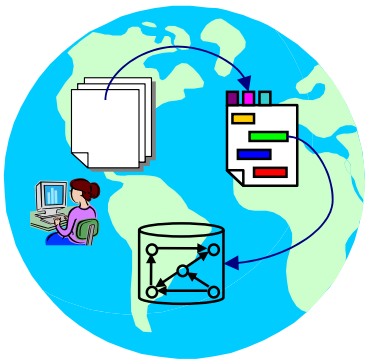


Fig. 2. Distributed Heterogeneous Knowledge

Based on the above arguments we propose decoupled representation and management of the documents, the metadata (annotations) and the formal knowledge (ontologies and instance data) as depicted on Fig. 2.

We will extremely shortly advocate the appropriateness of using ontology for defining the entity types – those are the only wide accepted paradigm for management of open, sharable, and reusable knowledge. According our view, light-weight ontology (poor on axioms) is sufficient for simple definition of the entity classes, their appropriate attributes, and relations. In the same time it allows more efficient and scalable management of the knowledge (compared to the heavy-weight semantic approaches.)

According to the analysis of ontology and knowledge representation languages and formats in [12] and other authors it becomes evident that there is no much consensus beyond RDF(S), see [4]. The latter is well established in the Semantic Web community as a knowledge representation and interchange language. The rich diversity of RDF(S) repositories, APIs and tools, forms a mature environment for development of systems grounded in RDF(S) representation of their ontological and knowledge resources. Because of the common acceptance of RDF(S) in the Semantic Web community, it would be easy to reuse the ontology and KB, as well as enrich them with domain-specific extensions. The new OWL (see [10]) standard offers clear, relatively consensual and backward-compatible path beyond RDF(S), but still lacks tool support. Our experience shows (see the section on KIM) that for the basic purposes of light-weight ontology definition and entity description, RDF(S) provides sufficient basic expressiveness. The most critical nice-to-have primitives (equality, transitive and symmetric relations, etc.) are well covered in OWL Lite – the simplest first level of OWL. So, we suggest that RDF(S) is used in a way which allows easy extension towards OWL[6] – this means avoiding primitives not included in the OWL schema.

3 The KIM platform

The KIM platform[7] provides semantic annotation, indexing, and retrieval services and infrastructure. The most important differences between KIM and other systems and approaches are that it performs semantic annotation and provides services based on the results. To do this in a consistent fashion, it performs information extraction based on an ontology and a massive knowledge base.

The traditional flat NE type sets consist of several general types (such as Organization, Person, Date, Location, Percent, Money). Although these represent the most important domain-independent NE types, still the entities with same type are dividable in more specific classes from the average educated human (e.g. public companies, sport teams, and syndicates are all organizations). We identified an inter-domain NE type hierarchy from a corpus of general news and integrated it in the KIM Ontology (KIMO). The ontology contains definitions of entities, relations, as well as a branch of lexical resource types (e.g. Title, PersonFirstName, DayOfWeek, etc.). The semantic descriptions of entities and relations between them are kept in a knowledge base (KB) encoded in the KIM ontology and residing in the same semantic repository. Thus KIM provides for each entity reference in the text (i) a link (URI) to the most

[6] http://www.w3.org/2002/07/owl

[7] Knowledge and Information Management Platform, see http://www.ontotext.com/kim

specific class in the ontology and (ii) a link to the specific instance in the KB. Each extracted NE is linked to its specific type information (thus Arabian Sea would be identified as **Sea**, instead of the traditional – **Location**). Also each NE is linked to an individual in the KB and the associated semantic description (attributes and relations of the entity). The KB has been pre-populated with entities of general importance, and is iteratively enriched with entity individuals and relations as a result of the IE process. Thus the extracted named entities could be further used for semantic indexing and retrieval of content with respect to entity instance and type. Thus allowing the satisfaction of requests that inquire for documents which refer entities described with type, name, and attribute restrictions, as well as the expected relations between these entities (e.g. look for a **Sea** that is a **subRegionOf** the Indian Ocean).

The information extraction process in KIM is based on the GATE platform. Few generic NLP components for tokenization, part-of-speech tagging, and others, have been directly reused by KIM. GATE's pattern-matching grammars have been modified to handle entity class information and allow generalization of the rules (e.g. specifying a pattern consisting of all **Location**s that are **subRegionOf** a **Country**, instead of specifying the concrete types of all the possible location sub-classes – **City, Province, CapitalCity**, etc.) The KIM gazetteer lookup component looks up entities and lexical resources by their aliases. The aliases present entity names or keys (suffixes, context words) and serve as clues for the pattern-matching grammar NER process. As a part of the KIM platform, the KIM IE is open towards the semantic repository that keeps the ontology and the KB, and depends on these for initialization of its processing components. Finally the IE identifies the instance information for each known NE in the text, and adds the new entities with their semantic descriptions and relations to the KB. Thus as a result each NE reference is linked to its type and its individual semantic description.

For the end-user, the KIM IE functionality is straightforward and simple – requesting annotation from a browser plug-in, which highlights the entities in the current content and generates a hyperlink used for further exploration of the available knowledge for the entity (as shown on Fig. 3). Various access methods are also available – entity pattern search, entity lookup, keyword and document attribute search. There is also an opportunity to create a composite query consisting of atomic searches of the above types.

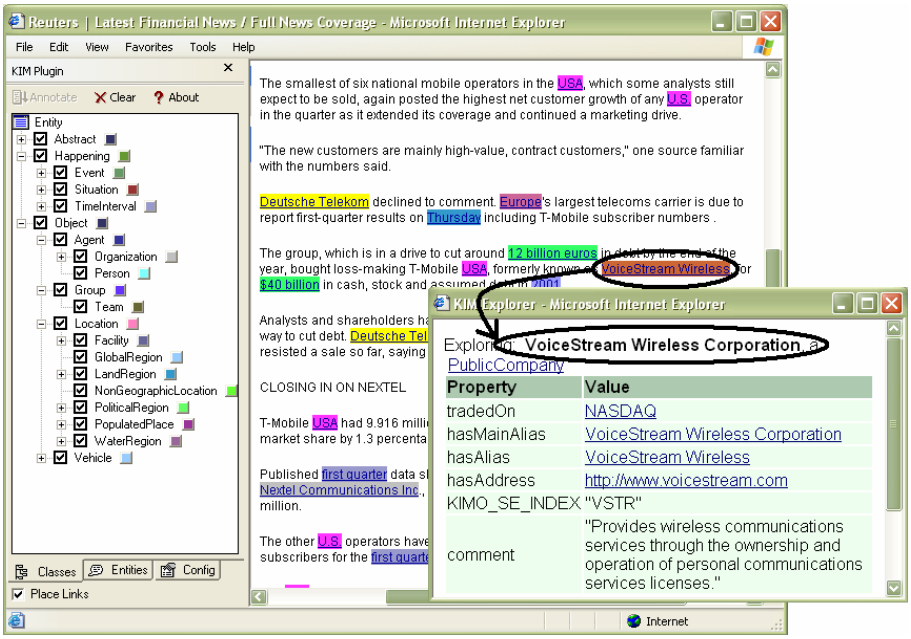


Fig. 3. KIM Plug-In, semantically annotated content and KB Explorer (on the front)

3.1. KIM Architecture

The KIM platform consists of KIM Ontology (KIMO) [8], knowledge base, KIM Server (with API for remote access, embedding, and integration), and front-ends (browser plug-in for Internet Explorer, KIM web user interface with various access methods, and Knowledge Explorer for KB navigation). The KIM API provides semantic annotation, indexing and retrieval services and infrastructure. KIM ontologies and knowledge bases are kept in semantic repositories based on cutting edge Semantic Web technology and standards, including RDF(S) repositories (SESAME[9] [5]), and ontology middleware[10] [15]. KIM provides a mature infrastructure for scalable and customizable information extraction, as well as annotation and document management, based on GATE [8]. The Lucene[11] information retrieval engine has been adapted to index documents by entity types and measure relevance according entities, along with tokens and stems. It is important to mention that KIM, as a software platform, is domain and task independent as are GATE, SESAME and Lucene. The KIM Architecture diagram is depicted on Fig. 4.

---

[8] http://www.ontotext.com/kim/2003/03/kimo.rdfs
[9] http://sesame.aidministrator.nl/, RDF(S) repository by Aidministrator b.v.
[10] OMM, http://www.ontotext.com/omm. Ontology Middleware Module is an enterprise back-end for formal knowledge management.
[11] Lucene, http://jakarta.apache.org/lucene/, high performance full text search engine

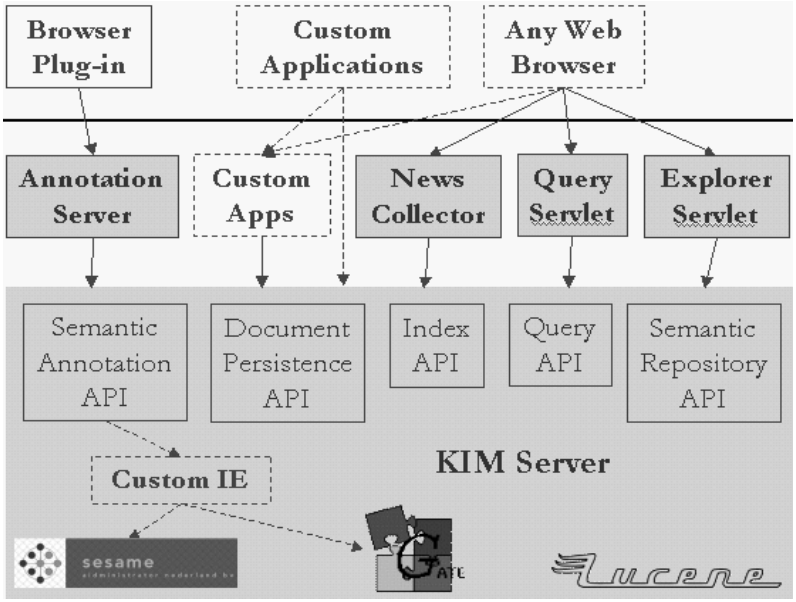


Fig. 4. KIM architecture – maybe change to include the KIM Web UI

3.2 KIM Ontology

The KIM ontology (KIMO) is a simplistic upper-level ontology starting with some basic philosophic distinctions between entity types (such as **`Object`**s - truly existing entities as locations and agents, **`Happening`**s - defining events and situations, and **`Abstract`**ions that are neither objects, neither happenings). Further on the ontology goes in more details, specifying real-world entity types of general importance (meetings, military conflicts, employment positions, commercial, government and other organizations, people, different types of locations, etc.). Also the characteristic attributes and relations for the featured entity types, are defined (e.g. **`subRegionOf`** property for **`Location`**s, **`hasPosition`** for **`Person`**s, **`locatedIn`** for organizations, etc.) Having this ontology as basis, one could add domain-specific extensions to it easily, to profile the semantic annotation for concrete applications. The integration of more than one domain-specific extension in a single application would not be possible without the intermediate role played by the upper-level ontology.

The KIM ontology (KIMO)[12] consists of 250 general entity types and 100 entity relations. The top classes are **`Entity`**, **`EntitySource`** and **`LexicalResource`**. The **`Entity`** class is further specialized into **`Object`**, **`Abstract`** and **`Happening`**. The top

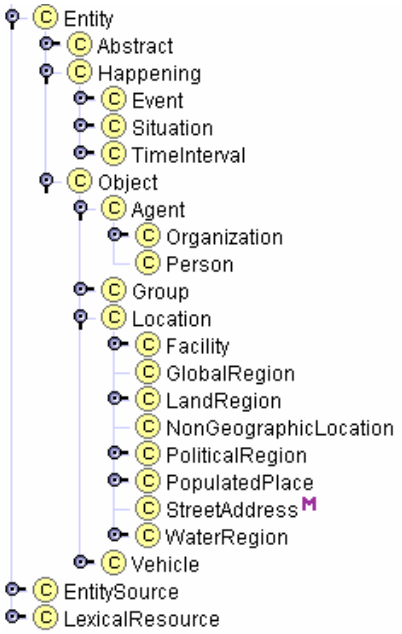
[12] http://www.ontotext.com/kim/2003/03/kimo.rdfs

Entities could be seen in the type hierarchy of the KIM plug-in on Fig. 3, and separately on Fig.5.
The **LexicalResource** branch is dedicated to encoding various data aiding the IE process, such as company suffixes (AG, Ltd.), person first names, etc. (depicted on Fig.5)

An important sub class of this branch is **Alias**, representing the alternative names for an **Entity** (see Fig. 7). The **hasAlias** relation is used to link an **Entity** to its alternative names. The official name of an entity is referred by the **hasMainAlias** property.

The instances of the **EntitySource** class are used to separate the trusted (pre-populated) information in the KB from the automatically extracted. This is indicated by the **generatedBy** property of the entity individuals.

The distribution of the most commonly referred entity types varies greatly from domain to domain (e.g. in a news corpus, the locations would be a much higher percentage from all entity annotations, than in an email corpus.) As researched in [18], despite the difference of type distributions, there are several general entity types that appear in all corpuses – **Person, Location, Organization, Money** (amount), **Dates**, etc. Further the ontology defines more specific entity types (e.g. **Mountain**, as a more specific type of **Location**.) The extent of specialization of the ontology is determined on the basis of research of the entity types in a corpus of general news (incl. political, sport, and financial, etc.)



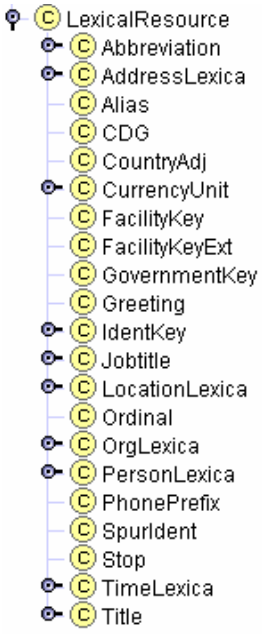


Fig. 5. The top of KIMO class hierarchy with expanded Entity branch. (on the left)
Fig. 6. The Lexical Resources top class hierarchy.

3.3 KIM Knowledge Base

The KIM KB represents a projection of the world, according to the domain that it is applied to. Our experiments are primarily in the field of international news. The specifics about this domain is that it covers the most well known and important entities in the world. KIM keeps the semantic descriptions of entities in the KIM KB, which is repeatedly enriched with recognized entities and relations. The entity descriptions are being stored in the same RDF(S) repository as the KIM ontology. Each entity has information about its specific type, aliases (incl. a main alias, expressing the (most probable) official name), attributes (e.g. **`Latitude`** of a **`Location`**), and relations (e.g. a **`Location subRegionOf Location`**). A simplistic schema of the entity representation is depicted on Fig. 7, where one could see the instance with its type and one alias.

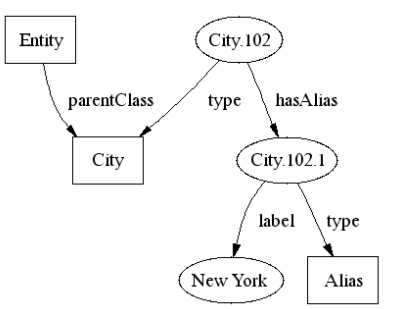


Fig. 7. Simplified view of the entity description

No matter how sophisticated the automatic IE process is, still one needs a starting KB to represent the entities that are considered important in the respective domain. This plenty of information should be carefully filtered in order to provide minimal, but representative coverage of the entities of general importance. There is no formal definition of the importance of an entity. However, we suggest that as important should be considered the entities that are well known to the wide public. Later on the importance of an entity could be represented through various ranking weights mostly derived statistically.

Pre-population of KIM KB.
KIM KB has been pre-populated with entities of general importance, that allow enough clues for the IE process to perform well on inter-domain web content. It consists of about 80,000 entities with more than 120,000 aliases. Various relations between entities are also predefined (like position of a person in an organization or company's allocation.)

The entities needed from the KB population are available on the web in the form of online encyclopedias, public servers, directories and gazetteers. For example the geographic locations and relations between them could be extracted [16] from NIMA's[13] GEOnet Names Server (GNS)[14], The Geographic Names Information System (GNIS)[15] data from the U.S. Geological Survey (UGCS), The Alexandria Digital Library (ADL) gazetteer[16], or other public geographic names server. The

[13] National Imagery and Mapping Agency of the US
[14] http://www.nima.mil/gns/html/
[15] http://geonames.usgs.gov/gnisform.html
[16] http://www.alexandria.ucsb.edu/

instances of important organization (and their officials) could be retrieved from the public directories of the biggest web portals, from other public servers, or in the form of compiled gazetteers.

KIM KB keeps the entity descriptions of frequently mentioned geographic resources. These entities have attributes and relations that depict their actual positioning and co-positioning in the physical world (such as **`longitude`**, **`latitude`**, **`subRegionOf`**). The GNS (GEOnet Names Server) has been used to extract the instances of the Location class. One of the most important relation types is **`subRegionOf`**, carrying the meaning that a region is a part of another one (e.g. **`Country subRegionOf Continent`**.) In its current state the KIM KB contains about 50,000 locations, including continents, global regions, 282 countries with their capitals, 4,700 cities (including all the cities with population over 100,000), mountains, big rivers, oceans, seas, and even oil fields. Each location has geographic coordinates and several aliases (usually including English, French and sometimes the local transcription of the location name) as well as co-positioning relations (e.g. **`subRegionOf`**.)

In the sources mentioned, the importance of the entities is not presented in an explicit form, and often there are even no clues for distinctions by this criterion (e.g. England and Scotland are listed in GNS alongside 40 other UK areas). On the other hand, some sources have inherent global importance specification of the contained entities (e.g. UN's list of cities with population over 100,000), but lack detailed attributes and relations, and cannot be used by themselves. The instances listed in such repositories are matched against exhaustive resources (e.g. GNS) and thus the significant entities are let through the filter, retaining their complementary disposition features (spatial attributes, and **`subRegionOf`** relations.)

The organizations with highest general importance also have been pre-populated in the KB. Including the biggest world organizations (such as UN, NATO, OPEC), over 7,900 companies, and 140 stock exchanges for a total of 8,400 organization instances. For the public companies (counting 5000 entities) there are 5500 position relations of managing personnel. The organizations also have locatedIn relations to the corresponding country instances. The additionally imported information about the companies consists of short description, URL, reference to an industry sector, reported sales, net income, and number of employees. The company data came from various sources, mostly per-country lists of registered companies. The company data is verified to contain all the publicly traded companies listed in the Google directory[17], Hoovers Online[18] and is currently being re-evaluated and enhanced with other important companies, according to the classifications of Forbes[19], Fortune magazines[20], and the European business directory[21].

Famous people (e.g. government officials, public company managing personnel) and some specific organizations (e.g. TV companies), have been also imported in the KB.

---

[17] http://directory.google.com/Top/Business/Major_Companies/Publicly_Traded/
[18] www.hoovers.com
[19] www.forbes.com
[20] www.fortune.com
[21] www.europages.net

In order to enable the IE process to recognize new entities and relations that are not a part of the KB, a collection of lexical resources is also presented in the KB. It covers organization suffixes, person names, time lexica, currency prefixes and others, serving as clues for the NER process.

Ensuring the quality of the KB content, is not trivial and could not be performed manually (having more than 80,000 pre-populated entities, the manual approach will simply not scale). The KIM KB is iteratively verified against an independently built KB of entities and relations collected manually from various web sources.

4 Semantic Information Extraction

The essence of the KIM IE approach is the recognition of named entities (NE) with respect to formal upper-level ontology (KIMO). The NE annotations are typed with respect to the entity classes in the ontology. The entity instances all bear unique identifiers that allow the annotations to be linked to the exact individual in the KB. The IE involved in KIM is currently concentrated mostly on the NER task, which is considered a step-stone for further attribute, relation, event, and scenario extraction. In order to identify the references of entity relations in the content, one should first have identified the entities. Usually the entity references are associated with a NE type, such as **`Location`**, **`Person`**, etc. More and more hierarchical NE type sets appear (f.e. [22]), especially for domain-specific applications. This is due to the need for finer grained specification and identification of world concepts. For example, it would be natural for an IE application performing company intelligence to keep more specialized sub-classes of **`Organization`** (e.g. such as **`PublicCompany`**). A NE type taxonomy however brings in a new level of complexity and (as discussed in section 5) sets new challenges for the evaluation of the performance, since the traditional Precision/Recall metrics are not directly applicable.

The IE process presented here uses light-weight ontology (KIMO) defining the entity types (called classes in the ontology slang.) In addition to the hierarchical ordering, each class is coupled with its appropriate attributes. The relation types are also defined with their domain and range restrictions. Actually, the basic ontology language used (RDFS) considers both the relations and attributes as properties, which can also be ordered in a hierarchy. Further, the ontology also has a branch of lexical resource classes (section 3.2). Given the ontology, the entities in the text could be linked to their type, which is also feasible with just a type taxonomy. However we would like to go further, and identify not only the type of the NE but also keep its semantic description and extend it with the IE process. Thus the NE references in the text are linked to an entity individual in the KB (section 3.3). The accessibility of the semantic descriptions of entities in the KB would allow the IE process to later base on attributes and relations as clues for recognition and disambiguation. For example, if a Person appears along with a Company in the content, and there are two companies that have the mentioned alias we have ambiguity. A possible approach would be to check whether the Person has some relations with one of the companies (e.g. working in it), and if so, the related Company to be chosen as a better candidate and associated with the NE reference in the content.

It is important to mention the opportunities that such IE would reveal for the access methods. Indexing (with customized Lucene) over the entity references in the text allows later on to perform IR with respect to entities. Thus one could specify the entities that are expected to be part of the result set of documents, with attribute and name restrictions (e.g. a **`Person`** which name ends with 'Alabama'). To solve this task we apply the semantic restrictions over the entities in the KB. Then the documents referring the resulting entities are being returned with ranking according to NEs. Even more one could specify a pattern of entities and relations between them, and restrict the entities by attributes, name and type.

KIM IE is based on the GATE framework, which has proved its maturity, extensibility and task independency for IE and other NL applications. We have reused much of GATE's document management functionality, and generic NLP components as its *Tokenizer, Part-of-Speech Tagger*, and *Sentence Splitter*. These processing layers are provided by the GATE platform, along with pattern-matching grammars, NE coreference and others, as standardized building bricks for easy construction of sophisticated IE applications.

For our purposes we changed the grammar components to handle entity class information and match rules according to it. The grammar rules are now based on the ontology classes, rather than on a flat set of NE types. This allows much more flexibility in the creation of NER rules at the most appropriate level of generality, giving both the opportunity to generalize and handle more specific NE types. A rule trying to extract relation between an organization and its point of presence can be specified at the level of the most general classes it applies to (**`Organization`** and **`Location`**) and still match a patterns with much more specific information (say, a radio station located in a county). On the other hand, instead of referring to all locations we could prefer to have rules that are especially applicable for **`Countries`**, **`Cities`**, or **`Seas`**.

The *Semantic Gazetteer* lookup component is based on the entities and lexical resources in the KB, rather than on file lists of aliases. Along with it all the reused components have been opened towards the semantic repository. For example the NE coreference module, in addition to the traditional ortho-matching techniques, handles the instance information of NE annotations and matches them according to it., as well as the traditional substring transformation matching. The *Semantic Gazetteer*, the simple disambiguation and annotation filtering components, as well as the final KB enrichment layer have been developed from scratch. These are not innate to a traditional NER and are inquired by the specifics of the Semantic IE, which takes care of the identification of NE references with respect to the ontology and KB.

The IE component flow diagram (Fig. 8) displays the sequential processing of content to the point where semantic annotations of NE are produced over it. The semantic repository is also displayed and linked with the ontology and KB aware components. The semantically-aware modules are presented in sub-sections below.

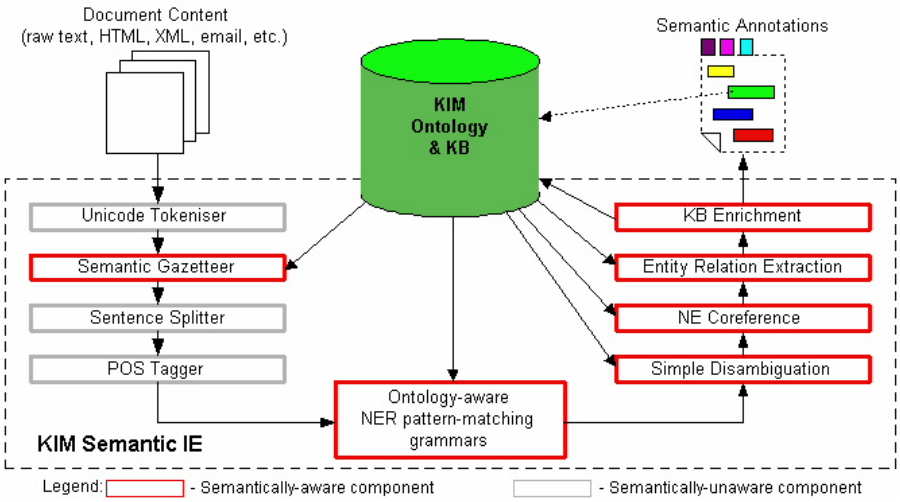


Fig. 8. KIM Semantic IE flow diagram.

4.1 Semantic Gazetteer

In the *Semantic Gazetteer* the lists of a traditional text-lookup component have been exchanged with a knowledge base that keeps the entities with their aliases and descriptions, as well as the lexical resources (such as possible male person first names). These are used to initialize the *Semantic Gazetteer* component, which keeps the various aliases and their type and instance references (as URIs). Upon occurrence of a known lexical resource or entity alias in the text (f.e. *Monday, John, GMT*, etc.), the *Semantic Gazetteer* generates a temporal annotation with a link to a class in the ontology (f.e. *Monday* will be linked to the KIM ontology class **`http://www.ontotext.com/kim/kimo.rdfs#DayOfWeek`**). Even more, the aliases of entities in the text are linked to the specific instances they refer to (f.e. *California* will be linked to the instance **`http://www.ontotext.com/kim/kimo.rdfs#Province.4188`**).

Since, many entities share aliases (f.e. *New York* is both a **`Province`** and a **`City`**) it often happens that one NE reference in the text is associated with several possible types and instances. At this phase we make sure all the equivalent possibilities are generated as annotations. Later on simple disambiguation techniques (section 4.4) are applied to filter some of the alternative annotations.

Although the KB contains both pre-populated and automatically recognized entities, only the former are used in the lookup process. The entities extracted from the processed content are not considered, and thus possible recognition mistakes are not reused as evidences. Let's consider we have previously extracted that within a given context the alias '*John*' referred to an entity with main alias (official name) '*John Smith*' and this entity with its semantic description has been added to the KB

and to the *Semantic Gazetteer* model. If the *Semantic Gazetteer* considered the recognized entities, the next time that '*John*' appears in the content it will be linked to the '*John Smith*' entity individual, and to many others with the same first name. But since the reference '*John*' doesn't really give a clue that one of the recognized entities with this first name is mentioned, the extracted information should be used with caution.

This phase is the entry-point for association of annotations in the text with a class in the ontology, and (for the entities) an instance in the KB. From here on the temporal annotations bare these semantic links, upon which the rest of the IE components base their processing.

4.2 Ontology-Aware Pattern-Matching Grammars

Pattern-matching grammars have proven to be applicable for various NLP tasks and also have traditionally been used for IE and NER. A grammar processor called JAPE [9] is a part of the GATE platform, and allows the specification of rules that fire on patterns of annotations. Thus one could specify actions and transformations that would take place if the rule is fired from a pattern in the content. We have modified the JAPE processor to handle class information and match patterns of annotations according to it. The NE grammars are based on the ones used in ANNIE[22] within the GATE project. In the modified grammars the definition of a rule goes through specification of the class restrictions for the entities in the pattern. The matching process uses the ontology to determine whether the candidate annotation has the same class as (or a sub-class of) the class in the pattern. Thus one could specify a pattern referring to a more general class (e.g. **`Organization`**), allowing all of its sub-classes (e.g. commercial, educational, religious and other organizations) to fire the grammar rule.

The pattern matching grammars are initially used to determine the entities within the processed content. At this point the suggested (by the *Semantic Gazetteer*) candidates for entities are evaluated. Some of them are considered credible and are transformed to final NE annotations. These inherit the type and instance information from the lookup annotations generated by the gazetteer. Other NE annotations are constructed by the grammar processor according to patterns in the content. These annotations have an entity type, but lack the instance information since they have not yet been associated with an existing KB individual. An example for identification of entities missing in the KB is using location/organization pre/post keys - "*River* Thames", "Mitsubishi *Corporation*", etc. Some context-based clues are also considered, such as "*in*" followed by Token-with-first-uppercase testifying that the latter is a **`Location`** (e.g. *in Kyoto*).

Later on, template relations extraction takes place, identifying some relations that the entities manifest in the content (determining the place where an organization is located; determining people's positions in organizations, f.e. *the CEO of NorthernStar, Mr. Yamamoto*).

---

[22] ANNIE, open-source, robust Information Extraction (IE) system based on pattern-matching grammars realized with finite state algorithms. http://gate.ac.uk/sale/tao/index.html#annie

4.3 Orthographic NE Coreference

The NER process continues with orthographic NE coreference component (see [2] and [11] for more on NE coreference within GATE), that generates lists of matching entity annotations within one type, according to their text representation (e.g. names like Mr. Malkovich and John Malkovich are usually referring the same entity individual within given context).

We have extended the coreference module to take into account the instance information of the recognized entities, thus enabling different string representations of an entity to be matched if they are aliases of one and the same KB individual. Without the instance data, names like Beijing and Pekin could not be matched only on the basis of orthographic coreference algorithms. The result of the coreference component is that groups of matching entities are identified. Later on these groups are used to determine the instance information and the aliases of new entities.

4.4 Simple Disambiguation

Potentially there are multiple entity-aliases in the KB that are equivalent to a NE reference in the text. For such references the *Semantic Gazetteer* generates multiple alternative annotations. Thus the over-generation of semantic annotations is rooted in the richness of the KB and the phenomena of naming different things with same names (e.g. Moscow being a **`CountryCapital`** and a **`City`** in US). At the level of the NER during the gazetteer lookup phase it is impossible to disambiguate because of the lack of clues (i.e. the gazetteer layer does not use evidence from other components, but the raw content itself). Later on simple disambiguation techniques take place in the pattern-matching grammars phase. For example, ambiguity between **`Person`** and **`Organization`** (e.g. "U.S. Navy"), would normally be recognized as a person name from the pattern: *two initials + first uppercas*, but in this case the initials match a location alias. Another problem is the occurrence of locations in person names, e.g."*Jack London*" (disambiguated because in the KB, "*Jack*" is known as a person first name).

Another class of ambiguities is the appearance of two annotations with different class and instance information over the same entity reference (*New York* being a **`Province`** and a **`City`**). Currently disambiguation of such annotations is not performed and this is subject of future work. For example, the context could be scanned for entities related to the ambiguous ones and thus relevance of the alternative entities to the content could be evaluated. For instance, if *Moscow* is used along with *Russia* its relevance is higher than the relevance of the alternative American city. We would also experiment with other approaches towards disambiguation of named-entity references. Adaptation of HMM learner, that has already successfully been used for non-semantic disambiguation is one of the first ideas. We would also like to experiment with techniques similar to those used for word-sense disambiguation (namely, lexical-chaining) and "symbolic" context management.

Beside the disambiguation in the grammar rules, a thin annotation filtering layer is used. More than one overlapping entity annotations (with same types) could be

recognized over the same part of the content. This is due to alternative patterns that fire the same rule or multiple trusted entities with the same alias. For example a person title (*Mr.*) followed by a looked up person candidate (e.g. *John Malkovich*), could match the left hand side of a rule, that also has an alternative firing pattern to match person titles followed by a token with upper-cased first letter (instead of looking for temporary person annotations as in the first pattern). As a result of the filtering only the annotations with distinct instance data are admitted - e.g. *New York* would be recognized both as a city and as a province, thus allowing later context-based disambiguation to determine the correct individual.

4.5 KB Enrichment

The last phase is not part of the standard IE systems, since it is related to the KB enrichment with new entity instances and relations. The newly recognized entity annotations lack instance information and are still not linked to the KB. However these entity annotations could represent entities that are in the recognized part of the KB. The first step is to match the entity annotations by their class information and string representation against the set of recognized entities. If a matching entity individual is found, the annotation acquires its instance identifier. Otherwise a new entity individual is constructed and added to the KB along with its aliases derived from the list of matching entities (if such).

At this point all generated named entity annotations are linked to the ontology (via their type information) and to the KB (via their specific instance). The relation annotations generated by the template relation extraction grammars, are used to generate the according entity relations in the KB (e.g. person's positions; spatial positioning information for organizations, etc.).

This finalizes the IE process, having as a result named entity annotations linked to their semantic descriptions in the KB.

5 Evaluation of KIM Named Entity Recognition

Along with the enrichment of the KB and the evolution of the IE process, we repeatedly evaluate the NER performance of KIM. This is needed to detect in early phases erroneous processing components or data. In order to test KIM NER most correctly it should be evaluated versus a corpus annotated with the specific type information. Such a metric however is not trivial and is subject of future work.

To measure the NER performance of KIM IE we have modified the GATE Corpus Benchmark Tool (CBT). CBT compares sets of annotations (key and response set) and calculates Precision, Recall, and F-measure. The metrics are presented separately for each document and combined for the final result. We also use the CBT to evaluate two sequential versions of the KIM platform against a human annotated corpus, thus determining the changes of the performance from version to version (regression testing).

The KIM NER performance has been evaluated, using CBT, against a corpus, human annotated with named entities. The evaluation corpus contains 100 documents

of news articles from UK media sources (Financial Times, Independent, and Guardian). The corpus is annotated with the traditional flat NE types used by most of the NER systems (**`Location, Organization, Person, Date, Percent,`** and **`Money`**). Despite the fact that KIM provides more specific type information, it is still possible to test it against the human annotated corpus (because something that is a **`Mountain`** is also a **`Location`**). In Table 1 we present the Precision, Recall and F-Measure of the automatically annotated corpus versus the human annotated one. These metrics are about the correctness of the KIM named entity recognition process in terms of general NE types, on the flat level of abstraction in standard NER systems.

Annotation Type	Precision	Recall	F-Measure
Date	0.92	0.83	0.87
Person	0.86	0.88	0.87
Organization	0.79	0.65	0.71
Location	0.87	0.92	0.90
Percent	1	1	1
Money	1	1	1
Total	0.86	0.84	0.85

Table 1. Evaluation of KIM NER wrt general NE types.

6 Related Work

Significant amount of research on IE has been performed in various projects related to GATE (see [17], [2], [7] [8] [9], [11], [18]). GATE provides tools such as tokenizers, part-of-speech taggers, gazetteer lookup components, pattern-matching grammars, coreference resolution tools and others that aid the construction of various NLP and especially IE applications. GATE is also a framework for content and annotation management. KIM's IE and content management is grounded in the GATE framework, and opens it towards Semantic Web knowledge representation and management technologies.

For some time now it has been obvious that the several general NE types used by the IE systems are not specific enough for many applications, that there are much more categories that matter. NE type hierarchies design has been discussed in [22].

Semantic annotation of documents with respect to ontology and entity knowledge base is discussed in [6] and [14] – although presenting interesting and ambitious approaches, these do not discuss usage of information extraction for automatic annotation. The focus of Annotea [14] is manual semantic annotation for authoring web content, while [6] targets the creation of a web-based open hypermedia linking service, backed by a conceptual model of document terminology.

Semantic annotation is used also in the S-CREAM project presented in [13] – the approach there is interesting with the heavy involvement of machine learning techniques for extraction of relations between the entities being annotated. Similar approach is taken also within the MnM project [21], where the semantic annotations

can be placed inline in the document content and refer to an ontology and KB server (WebOnto), accessible via standard API.

An interesting named entity indexing and question/answer system is presented in [19]. Here flat set of entity types is assigned to tokens and the annotations are incorporated in the content, in order to index by NE type later. Once indexed the content is queried via natural language questions, with NE tagging over the question used to determine the expected answer type (e.g. When have the United Nations been established; UN here would be tagged with _ORG, thus specifying that the expected answer type is organization.) This approach is also interesting because of its question/answer interface, allowing the users to specify their queries in NL sentences (with few limitations).

Experiments with the acquisition of spatial knowledge and its usage for IE have been described in [16].

Significant work on ontology and metadata infrastructure has been undertaken in the KAON project [3], which shares similarities with SESAME [5].

All the semantic annotation techniques referred above lack the usage of upper-level ontologies and critical mass of world knowledge to serve as a trusted and reusable basis for the automatic recognition and annotation, as in the approach presented in [1] and discussed here. Also the IE processes involved in related work do not link the NE reference in the text with a NE individual in the KB. Because of this unique feature the semantic description of the entity instance reveals its attributes, aliases, type, origin source, and relations with other individuals.

7 Conclusion and Future Work

We presented the Semantic IE approach embodied in the KIM Platform, with the involved technologies and resources.

Even linguistically simplistic, KIM platform provides a test bed and proofs number of hypothesis and design decisions:

- It's worth using almost-exhaustive entity knowledge (sort of super-gazetteers) for information extraction. The technology used (based on GATE) can manage the scale. Even without significant efforts on disambiguation, the precision drawbacks are acceptable for many applications;
- It is possible to adopt a traditional symbolic IE system to perform semantic annotations and thus provide its results in shape suitable for Semantic Web applications;
- A simple but efficient technique for entity-aware IR is demonstrated based on indexing over semantic annotations, which is an interesting example of IR engine taking benefit of the IE process.

The implementation is currently under development, so, preliminary results are reported. The evaluation work done until now does not provide enough evidence regarding the approach, technology, and resources being used. The major reason for this is that there are neither test data nor well developed metrics for semantic annotation and retrieval.

There are number of challenges for the Semantic IE which we will address in our future work:

- Develop (or adapt) evaluation metric which properly measures the performance of a semantic annotation system;
- Experiment different approaches towards disambiguation of NE references
- Make use of more advanced IE-techniques for identification of relations, analysis of events and situations, etc.
- The KIM Ontology and KB as well as the methodology and procedure for their sustainable maintenance and improvement will be subject of future research.

References

1. Bontcheva K., Kiryakov A., Cunningham H., Popov B., Dimitrov M.. *Semantic Web Enabled, Open Source Language Technology*. In proc. of EACL Workshop "Language Technology and the Semantic Web", NLPXML-2003, 13 April, 2003
2. Bontcheva K., Dimitrov M., Maynard D., Tablan V., Cunningham H., *Shallow Methods for Named Entity Coreference Resolution.* Chaînes de références et résolveurs d'anaphores, workshop TALN 2002, Nancy, France, 2002.
3. Bozsak E. et al. KAON - *Towards a large scale Semantic Web*, EC-Web 2002
4. Brickley D, Guha R.V., eds. *Resource Description Framework (RDF) Schemas,* W3C http://www.w3.org/TR/2000/CR-rdf-schema-20000327/
5. Broekstra J., Kampman A., and van Harmelen F. - Sesame: An architecture for storing and querying RDF data and schema information. In H. Lieberman D. Fensel, J. Hendler and W. Wahlster, editors, Semantics for the WWW. MIT Press, 2001
6. Carr L., Bechhofer S., Goble C., Hall W.. *Conceptual Linking: Ontology-based Open Hypermedia.* In The WWW10 Conference, Hong Kong, May, pp. 334-342.
7. Cunningham H., Information Extraction: a User Guide (revised version). Department of Computer Science, University of Sheffield, May, 1999
8. Cunningham H., Maynard D., Bontcheva K. and Tablan V., *GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications*. In Proc. of the 40th Anniversary Meeting of the Association for Computational Linguistics, 2002.
9. Cunningham, H. and Maynard D. and Tablan V., JAPE: a Java Annotation Patterns Engine (Second Edition). Technical report CS--00--10, Univ. of Sheffield, Department of Computer Science, 2000.
10. Dean M., Connolly D., van Harmelen, F., Hendler J., Horrocks I., McGuinness D., Patel-Schneider P., Stein L.A., *Web Ontology Language (OWL) Reference Version 1.0.* W3C Working Draft 12 Nov. 2002, http://www.w3.org/TR/2002/WD-owl-ref-20021112/
11. Dimitrov M., Bontcheva K., Cunningham H., Maynard D., *A Light-weight Approach to Coreference Resolution for Named Entities in Text*. Proceedings of the Fourth Discourse Anaphora and Anaphor Resolution Colloquium (DAARC) , Lisbon, September 2002.
12. Fensel D., *Ontology Language, v.2 (Welcome to OIL)* . Deliverable 2, On-To-Knowledge project, December 2001. http://www.ontoknowledge.org/downl/del2.pdf
13. Handschuh S., Staab St., Ciravegna F., *S-CREAM – Semi-automatic CREAtion of Metadata.*The 13th International Conference on Knowledge Engineering and Management (EKAW 2002), ed Gomez-Perez, A., Springer Verlag, 2002.

14. Kahan J., Koivunen M., Prud'Hommeaux E., Swick R.. *Annotea: An Open RDF Infrastructure for Shared Web Annotations*. In The WWW10 Conference, Hong Kong, May, pp. 623-632.
15. Kiryakov A., Simov K.Iv., Ognyanov D., *Ontology Middleware: Analysis and Design* Del. 38, On-To-Knowledge, March 2002. http://www.ontoknowledge.org/downl/del38.pdf
16. Manov D., Kiryakov A., Popov B., Bontcheva K., Maynard D., Cunningham H., *Experiments with geographic knowledge for information extraction,* NAACL-HLT 2003, Canada., Workshop on the Analysis of Geographic References, May 31 2003, Edmonton, Alberta.
17. Maynard D. ,Tablan D. ,Ursu C., Cunningham H., Wilks Y., Named Entity Recognition from Diverse Text Types. *Recent Advances in Natural Language Processing 2001 Conference*, Tzigov Chark, Bulgaria.
18. Maynard D., Tablan V., Bontcheva K., Cunningham H, and Wilks Y., *MUlti-Source Entity recognition – an Information Extraction System for Diverse Text Types*. Technical report CS--02--03, Univ. of Sheffield, Dep. of CS, 2003. http://gate.ac.uk/gate/doc/papers.html
19. Moldovan D., Mihalcea R.. *Document Indexing Using Named Entities.* In "Studies in Informatics and Control", Vol. 10, No. 1, March 2001.
20. van Ossenbruggen J., Hardman L., Rutledge L., *Hypermedia and the Semantic Web: A Research Agenda*. Journal of Digital information, volume 3 issue 1, May 2002.
21. Vargas-Vera M., Motta E., Domingue J., Lanzoni M., Stutt A. and Ciravegna F,, *MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup*, In Proc. Of EKAW 2002, ed Gomez-Perez, A., Springer Verlag, 2002.
22. Sekine S., Sudo K., Nobata Ch., *Extended Named Entity Hierarchy* (LREC 2002)
23. Pustejovsky J., Boguraev B., Verhagen, M., Buitelaar P., and Johnston M., *Semantic Indexing and Typed hyperlinking*. In Proceedings of the American Association for Artificial Intelligence Conference, Spring Symposium, NLP for WWW, 120-128. Stanford University, CA, 1997.