Facilitating Non-expert Users of the KYOTO Platform: the TMEKO Editing Protocol for Synset to Ontology Mappings

Roxane Segers[1], Piek Vossen[2]

[1] VU University Amsterdam, FEW, Department of Computer Science, De Boelelaan 1081A, 1081 HV Amsterdam, The Netherlands

[2] VU University Amsterdam, Department of Humanities, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands

E-mail: rh.segers@cs.vu.nl, p.vossen@let.vu.nl

Abstract

This paper presents the general architecture of the TMEKO protocol (Tutoring Methodology for Enriching the Kyoto Ontology) that guides non-expert users through the process of creating mappings from domain wordnet synsets to a shared ontology by answering natural language questions. TMEKO will be part of a Wiki-like community platform currently developed in the Kyoto project (http://www.kyoto-project.eu). The platform provides the architecture for ontology based fact mining to enable knowledge sharing across languages and cultures. A central part of the platform is the Wikyoto editing environment in which users can create their own domain wordnet for seven different languages and define relations to the central and shared ontology based on DOLCE. A substantial part of the mappings will involve important processes and qualities associated with the concept. Therefore, the TMEKO protocol provides specific interviews for creating complex mappings that go beyond subclass and equivalence relations. The Kyoto platform and the TMEKO protocol are developed and applied to the environment domain for seven different languages (English, Dutch, Italian, Spanish, Basque, Japanese and Chinese), but can easily be extended and adapted to other languages and domains.

1. Introduction

Experts have tremendous knowledge about their domain concepts but not the means for modeling this knowledge in a way that ensures reusability across languages and cultures. The need for reusing and exchanging knowledge is especially pressing within the environment domain since specific environmental issues are seldom restricted to single countries; a decrease in a migration bird population in a certain area might be related to changed hunting regulations in another part of the world.

Knowledge about environmental issues is stored in large amounts of document collections that are now only partially accessible through keyword search. The drawbacks of this approach are obvious: search results for a decrease in species in a certain area are limited, since only a few fragments with the specific keywords will be shown. Also, the keywords might appear in different places in a text and not be related at all. Relevant information in other languages is not easily accessible either. To ease the search for relevant information and to enhance information sharing between different languages, the Kyoto project is developing a community platform for modeling knowledge and finding facts across languages and cultures (Vossen et al., 2008). The platform operates as a Wiki and establishes semantic interoperability across languages for the environment domain by creating domain wordnets for seven languages that are interlinked through a shared DOLCE based ontology (Masolo et al., 2003).

Users are able to upload documents that will be processed in the Kyoto language processing pipeline that includes lemmatizing, syntactic parsing, creation of dependency trees, word sense disambiguation, named entity recognition and ontology tagging on word sense level. Each of these modules adds specific layers to the Kyoto Annotation Format (KAF) (Bosma et al., 2009), a LAF based text annotation format (Ide & Romary, 2003). The KAF annotated documents are the source for the term extractor (Tybot) that extracts relevant concepts from the texts in hierarchical structures (Bosma et al., 2010). The resulting termdatabases for each language are part of the Kyoto knowledge base that also comprises a SKOS converted Species2000 database [1], the seven generic wordnets, and the shared ontology. All these components (except the ontology) are presented to the users of the Wikyoto editing platform as an input for creating a domain wordnet.

The terms extracted from the documents and the species in the Species2000 database are disambiguated and partially aligned with synsets in the generic wordnets. In this way, each new synset in the domain wordnets hierarchy is linked to a synset in the generic wordnets by traversing the hierarchy. By this alignment, the existing mappings from the generic wordnets to the ontology can be used to apply the ontological distinctions to the domain terms. As such, the platform allows for continuous updating and modeling of the vocabulary by the people in the community, while their domain wordnets remain anchored to the generic wordnets.

Knowledge is added to the documents by creating domain wordnets and additional mappings from the domain synsets to the central ontology. Domain specific terms can then be recognized and annotated with their synset ID and according ontological information in the KAF. Ontological patterns that express domain knowledge on an abstract level can then be applied to the processed texts and find relevant information that can be lexicalized in various ways in documents from different sources or written in different languages. These patterns (Kybots) will mine facts from the documents and store these facts in a fact database. This fact database allows for semantic search and directs the user to the actual document where this information was found. These different components of the Kyoto architecture are presented in figure 1.

[1] http://www.sp2000.org

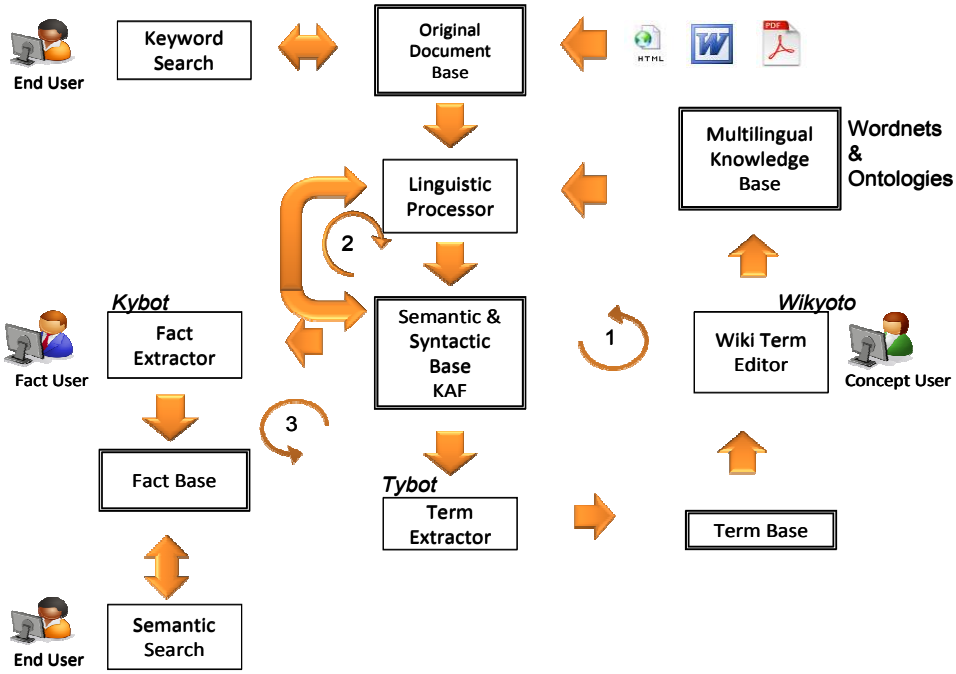

Figure1:Kyoto System Overview

Crosscultural and crosslingual information sharing is driven by the users as they model their knowledge by creating wordnets and ontology mappings. Each contribution in the domain wordnets results in finding more and specific information in the document collection and therefore provides a direct return of investment. The task of creating wordnets and ontology mappings however requires specific knowledge; this can be a bottleneck for those users of the system that have no background in linguistics or knowledge engineering. The TMEKO procedure is designed to facilitate users by generating natural language questions that guide the users through the editing process and hide the underlying ontological decisions as much as possible.

The remainder of this article is as follows: in section 2 we describe the ontology that forms the Kyoto knowledge base. In section 3 we provide the requirements for the design of TMEKO and in section 4 the protocol is shown with examples of the different modules and interviews involved. In section 5 we conclude on the TMEKO protocol and discuss some of the open issues and future work.

2. The KYOTO knowledge base

The Kyoto knowledge base consists of the following components:
-Generic wordnets for all seven languages, partially linked to the shared Kyoto ontology.
-Species2000 database[2], partially linked to the seven generic wordnets.
-Kyoto term collection for all seven languages, partially linked to the generic wordnets.
-Domain wordnets for all seven languages (in progress).
-Kyoto ontology, DOLCE based and extended to meet the requirements of the Kyoto project.
The next paragraphs describe the main characteristics of the Kyoto ontology that plays a key role in the TMEKO procedure.

2.1 Ontology: modeling choices

Currently, the Kyoto ontology contains 1133 classes and 332 object properties divided over a top level based on DOLCE[3], a midlevel based on noun Base Concepts in the English wordnet (Izquierdo et al., 2007), and the low level that represents relevant concepts that are selected by the domain experts. The scope of the ontology itself is relatively small: many concepts remain in the generic wordnets and in the different databases in the knowledge base that are aligned with the generic wordnets. For text mining in documents about environmental issues it is sufficient to define processes and roles on a relatively high level of abstraction. Extensive hierarchies of species have therefore not been added to the ontology.
The Species2000 database that was partially used to enrich the generic wordnets contains about 2.1 million species. It seems important for the environment domain to include all these species to the ontology, but there are two reasons for not adding these. A practical argument is that no inference system can load an ontology of that size, but more important is the observation that the learned terms from the document collection typically express roles of species and other objects (*endurants*), and not the species themselves. We find e.g. species in a role like *red list species*, *alien invasive species* and *mobile vector species*. The domain experts themselves are also specifically

[2] http://www.sp2000.org

[3] The Kyoto ontology includes Dolce Lite Plus and upper level ontology extensions.

interested in the roles and processes (*perdurants*) a species participates in, like what species are endangered by water pollution, or what birds indicate an improving biodiversity in some estuary. Adding species and defining axioms in the ontology will therefore not help the extraction of this kind of domain specific knowledge from the document collections.
Since the species in the Species2000 database are partially aligned with the different wordnets, one can still infer that *Vipera berus* is a hyponym of 'snake' and related to the Class Reptile and Animal in the ontology. A Kybot profile that searches for e.g. [Migration] of [Species] to a [Region][4] in the document collection will likewise be able to infer that *Vipera berus* is a snake species without this knowledge being expressed explicitly in the ontology.

2.2 Relations in the ontology

The ontology is the backbone of the Kyoto platform since it forms the formal interlingua between the domain wordnets in different languages. The necessity of this shared ontology can be explained by two examples that show how knowledge can be lexicalized quite differently in different languages. The plant species *Urtica Dioica* (stinging nettle) for instance has the role *waardplant* (a plant that serves as food for caterpillars) in Dutch. This plant and its specific function in the ecosystem are known in many countries but this specific role is not lexicalized in all languages. A concept that comes close to the Dutch *waardplant* is the English *host*; this term however is used as a role for different organisms and not restricted to plants. Another case of differences in domain knowledge are objects that are not shared amongst languages as the concept is simply not known. The Dutch term *wiel* for instance expresses a specific kind of pond that is found only close to dikes. The domain wordnet for e.g. Dutch will be organized differently and contain synsets that are absent in domain wordnets for English or Japanese. Formalized knowledge about these culture-specific concepts is obviously necessary to be able to share knowledge across cultures and expressed in different languages.
Another argument for having a shared ontology is that wordnet hierarchies do not always follow clear ontological distinctions as they model language in a somewhat intuitive way. An ontological metaproperty like rigidity does not play a role in the way wordnets are modeled, but this notion of rigidity is of great importance for useful inferences (Guarino & Welty, 2002; Gangemi et al., 2003). Rigid concepts represent properties that are essential to all of their instances, while non-rigid concepts represent properties that exist only contingently for some of their instances. For example, *snake* is a rigid concept but *prey* is not: each snake must always be a snake under all circumstances or else it ceases to exist. A prey, however, ceases to be a prey when it is not longer hunted. Clear ontological distinctions cannot always be applied in wordnets since the concepts that could express this distinction are not lexicalized. In the case of *prey*, *pet* or *predator* there is no lexicalized intermediate level between these and e.g. *animal* to express that a *prey* is not a kind of animal, but an animal in a role. As such, both rigid synsets like *snake* and a non-rigid synsets like *prey* will have the hypernym *animal* in the wordnet, but are related with different mappings to the ontology.

Figure 2 presents an overview of the different relations between synsets en between synsets and the ontology; the boxes represent ontology classes, the ovals represent synsets.
- Wordnet internal relations. The synsets in each domain wordnet are internally related with EuroWordnet relations (Vossen 1998) like HAS_HYPERNYM, HAS_MERONYM, HAS_ROLE, etc.
- Wordnet synset to ontology. The relations from a synset to the ontology are prefixed with 'sc_' standing for synset-to-concept. The mapping from synsets to the ontology is different for rigid and non-rigid synsets. For *rigid* synsets (e.g. snake, tree, pond) there are sc_subclassOf and sc_equivalenceOf relations between synsets and Endurant, Perdurant or Quality. For *non-rigids* (e.g. pet, prey, predator), we have an sc_domainOf between synsets and Endurants, and many relations for mapping non-rigids to important processes, states and qualities. In this figure there is a sc_playRole relation between the synset *prey* and two Roles.
The following relations are currently available for creating mappings:
-sc_instanceOf :
Humber sc_instanceOf Estuary
-sc_participantOf:
migratory bird sc_participantOf Migration
-sc_playRole:
migratory bird sc_playRole done-by
-sc_hasParticipant:
bird migration sc_hasParticipant Bird
-sc_hasRole:
bird migration sc_hasRole done-by
-sc_hasLocation:
river water sc_hasLocation River
-sc_hasPart:
motor vehicle sc_hasPart Engine
-sc_partOf
sea salt sc_partOf Sea
-sc_hasState:
clear water sc_hasState Clear
-sc_stateOf:
water clarity sc_stateOf Water
-sc_hasCoParticipant:
gas-powered vehicles sc_hasCoParticipant Gas
-sc_playCoRole:
gas-powered vehicles sc_playCoRole Resource
The TMEKO protocol enables the users to create these kind of complex mappings without prior knowledge about the ontology itself.
- Ontology internal relations. The ontology is organized as follows: Classes can have subclasses, Roles are related to Endurants with a playedBy relation; Perdurants have a hasRole relation to Roles, and Endurants are related to Perdurants with a participatesIn relation.

[4] This is a very simplified example of the actual Kybot profiles. Elaborated and operational versions of Kybot profiles are available on the Kyoto website: http://www.kyoto-project.eu.

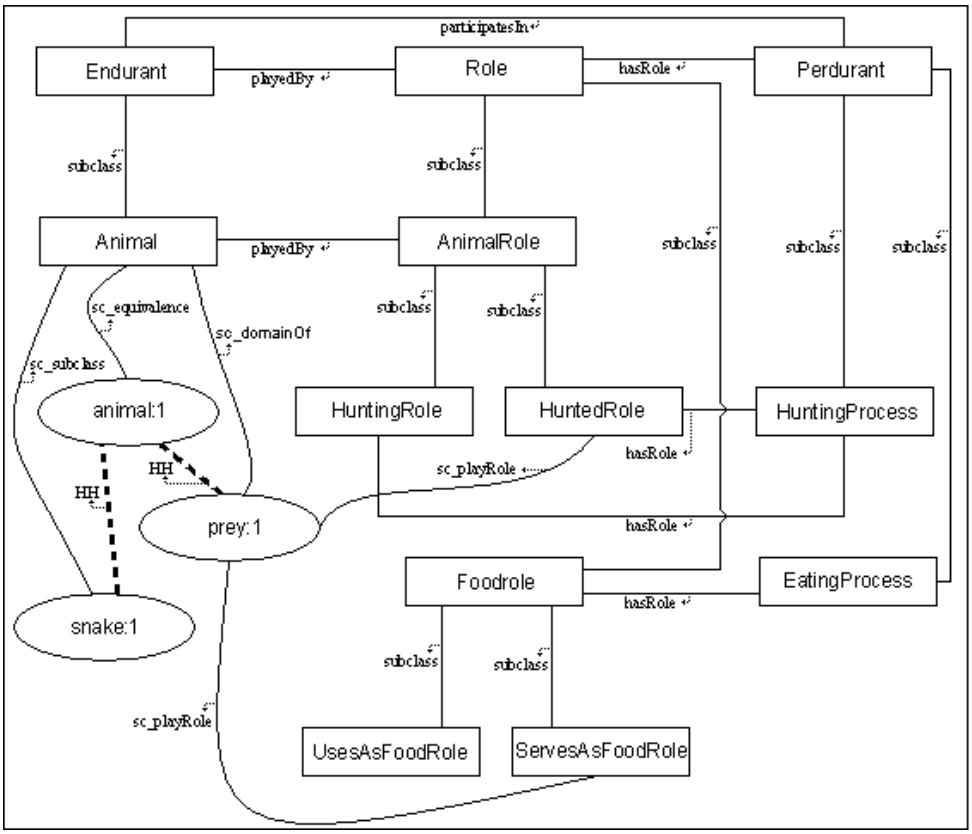

Figure 2: Overview of relations in the ontology, from synsets tot the ontology and from synset to synset

3. The TMEKO protocol

Ontology and wordnet editing and extension are usually carried out by knowledge engineering experts. In the Kyoto platform, non-expert users will create a domain wordnet and mappings from this wordnet to the shared ontology. A clear distinction is made between the expert mode and the non-expert mode in the overall editing process. Experts can edit and extend the ontology and they are allowed to directly create mappings between synsets and ontology classes. Non-experts will only extend the domain wordnets and create mappings from synsets to ontology classes. Extension and mapping will be done via the TMEKO procedure that helps the user in finding correct hypernym relations in the wordnets, and guides them through the process of creating mappings. The users will not see the ontology and they will also not be bothered with ontological issues or decisions. To make this possible, the TMEKO procedure is heavily supported by mined definitions and sets of natural language validation questions that are generated by the system.

The TMEKO protocol is inspired on the TMEO methodology (Tutoring Methodology for the Enrichment of Ontologies) that was created for the Italian Senso Comune project[5] (Oltramari&Vetere 2008). In this project non-expert users can extend a lexical resource for Italian that is mapped to an ontology based on DOLCE. The users only work at the lexical level and are guided by a QA system called TMEO for linguistic enrichment of the ontology. For any lemma that a user wants to add to the knowledge base, the system selects the most adequate ontology class and presents questions to the user. These questions are based on the distinctions in the DOLCE ontology. In TMEO, a lemma like 'glass' and its according gloss are presented to the user. Next, a superclass is selected and a set of questions is presented like '*can you count [glass]?*' and '*is [glass] produced/built by hand/machines?*'. The answer to each question then corresponds to an ontology class, and thus the meaning of the lemma is further specialized. We used the TMEO procedure as a starting point for designing a Kyoto version.

The resulting TMEKO protocol is a profound adaptation of the TMEO methodology. TMEKO differentiates between rigid and non-rigid synsets and enables the creation of complex mappings for non-rigid synsets to roles, processes and qualities by interviews that rely heavily on mined and processed definitions of the non-rigid concepts. The rigidity of the synsets is evaluated automatically at the beginning of each single editing process and needs not to be determined by the users.

Figure 3 gives an overview of the general TMEKO architecture. The procedure starts after the user has selected a synset or synset hierarchy from one of the databases in the knowledge base and adds it to a hypernym in the domain wordnet hierarchy. As stated, all databases in the knowledge base are partially linked to the generic wordnets. The hypernym chain will be traversed automatically to return the closest ontology class for the synset that is added by the user. As a result, the TMEKO procedure will start at the lowest possible level in the

[5] http://www.senso-comune.it/portale/

ontology.
The ontology class that is returned, defines at which point the TMEKO procedure will be started. If the selected class is a Perdurant, the user will be directed to interviews that find relations to e.g. the participants of the process.
If the selected class is an Endurant, the rigidity of the synset needs to be determined. After this point, TMEKO makes a distinction between rigid synsets for which a simple mapping is created, and non-rigid synsets for which the interviews are needed to provide complex mappings to the ontology. To determine the rigidity of synsets, a set of tools called Rudify was developed within the Kyoto project. Rudify provides a semi-automatic evaluation of the ontological meta-properties rigidity and non-rigidity based on lexical realizations of these meta-properties in natural language (Hicks & Herold 2009).
For rigid terms, no interviews will be generated as we expect that these synsets in the domain wordnets will be related with a sc_subclassOf relation to an ontology class. Each generic wordnet has a predefined set of Base Concepts like 'animal', 'river' and 'mineral' that are already mapped to the shared ontology with a sc_equivalenceOf relation. New domain specific synsets are most likely to be added below this level in the generic wordnets. For rigid synsets, this will automatically imply a sc_subclassOf relation to the selected ontology class.
For non-rigid synsets and synsets that denote a process, definitions will be mined from the document collection and through a Google search. Users can select the best definitions and the important qualities, processes and states associated with the synset. These selected words in the definitions will be associated with different ontological classes. Predefined templates will then generate simple yes/no questions that check how these classes are associated to the synset. The positive answers to these questions will generate the final complex mappings that are stored in the domain wordnets.

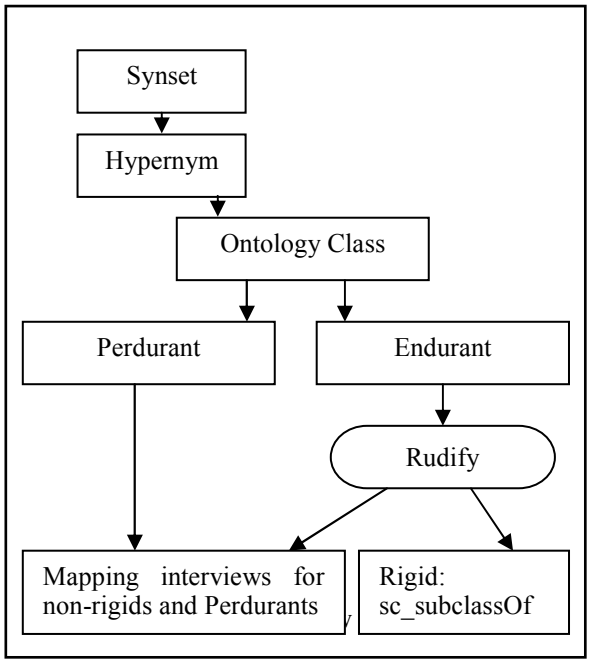

Fig. 3: overview of the general TMEKO architecture

4. The TMEKO interviews for creating complex mappings

This section shows the different steps and interviews involved in the TMEKO procedure for non-rigid synsets and synsets that denote processes. The interviews are elaborated for the non-rigid concept 'prey' and the process 'bird migration'.[6]

A. TMEKO for the non-rigid synset prey

1. Start of the TMEKO procedure
The user adds the synset 'prey' to the domain wordnet and relates it to the hypernym 'animal'. The hypernym chain is traversed and returns the Endurant ontology class Animal.

2. The Rudify tool generates a rigidity score for the concept
The performance of the Rudify tool is dependent on the frequency of a lexical representation in either the document collection or Google; therefore low frequency terms can have non-reliable rigidity scores. In these cases the Rudify output needs to be validated by the user.

2A. Validation questions for the Rudify output (optional interview)
The next validation questions are generated if Rudify would determine that 'prey' is non-rigid with a score below the treshold:

-Is something born or did it came in existence as a [*prey*] and will it always and necessary be a [*prey*]?
No: term is indeed non-rigid. Proceed in TMEKO.
Yes: possible wrong Rudify output. Another verification question is generated:
-When a [prey] stops being a [*prey*] will it no longer be in existence?
Yes: term is rigid. Creation of sc_subclassOf mapping.
No: term is non-rigid. Proceed in TMEKO.

3. Additional confusion interviews (optional)
An optional additional confusion interview is started. These questions are partly based on regular polysemy and can also be learnt from earlier errors and fixes by the users that are stored in the logfiles. These confusion interviews will therefore only be generated for specific classes in the ontology. In the case of the selected class Animal, the user can be unaware of the distinction between the class of countable animals (Animal) and the class of TaxonomicGroup. In this case, two validation questions are generated:

-Is [prey] expressing the *(Latin) scientific name* of a group of organisms?
Yes: (select TaxonomicGroup)
No: (Animal is the correct class)

[6] In the final interface for the TMEKO protocol, each step will have a help button that provides a short explanation and illustrating examples.

-Is [prey] an expression for an *organism* that you can touch, see, count or observe?
Yes: (Animal is the correct class)
No: (select TaxonomicGroup)

If the user denies both suggestions, he can opt for relating the synset to a different hypernym based on a list of suggestions, or ask for starting a full class based interview that goes top-down through the ontology.

4. Mining definitions
For the non-rigid synset prey, definitions are mined in the document collection and by a Google search. The used patterns are e.g. 'X is Y that', 'X is a part of Y that', X is a kind of Y that', 'X refers to'. For *prey*, the following definitions were found:

-*A prey is an animal that* is hunted by a predator.
-*A prey is an animal that* is being hunted or eaten by other animals.
-*A prey is an animal that* is hunted for food.

These definitions are parsed to generate KAF (Kyoto Annotation Format), including word sense disambiguation and term detection. The content words in the that-clause are automatically mapped to synsets and through the synsets to an ontology class. The user selects the processes and qualities that are of importance for the concept; the according classes for these synsets form the basis for creating relations. If necessary, the user can delete definitions or add one manually.

5. Presentation and selection of candidates
The candidate synsets that form the basis for creating mappings to the ontology are presented to the user.
The user can check the senses of the candidate synsets. If the WSD module selected the wrong synset, the user can select another one from a list of suggested synsets ranked by a confidence score. In the case of *prey* the following synsets are selected:

-animal:1, animate being:1, beast:1, brute:2, creature:1, fauna:2 *(a living organism characterized by voluntary movement)*.
Ontology class: Animal
-hunt:1, run:36, hunt down:1, track down:1 *(pursue for food or sport (as of wild animals))*.
Ontology class: HuntingProcess
-food:2, solid food:1 *(any solid substance (as opposed to liquid) that is used as a source of nourishment)*. (Ontology class Food)
-feed:6, eat:3 (*take in food; used for animals only)*.
Ontology class: EatingProcess

6. Create mappings to the ontology
The mapping of *prey* to the ontology class Animal is stored as an sc_domainOf relation. Each selected synset from the mined definitions is related to an ontology class. Relations of the selected processes, states and qualities are paraphrased in templates and presented to the user by a set of simple yes/no questions:

Is it true for [prey] that it is involved in hunting?
(yes/no) → (HuntingProcess)
Is it true for [prey] that it hunts something or someone?
(yes/no) → (HuntingRole)
Is it true for [prey] that it is hunted?
(yes/no) → (HuntedRole)
Is it true for [prey] that it is involved in eating?
(yes/no) → (EatingProcess)
Is it true for [prey] that it eats something/someone?
(yes/no) → (UsedAsFoodRole)
Is it true for [prey] that it is eaten?
(yes/no) → (ServesAsFoodRole)

From the positive answers to these questions, the following relations to the ontology will be stored in the domain wordnet:

prey:
sc_domainOf: Animal
sc_playRole: HuntedRole
sc_playRole: ServesAsFoodRole

From the definitions and interviews for the counterpart synset 'predator' the next mapping relations will be stored:

predator:
sc_domainOf: Animal
sc_playRole: HuntingRole
sc_playRole: UsesAsFoodRole

This type of mapping relation requires the presence of many and specific Roles related to Endurants and Perdurants in the ontology which are now only partially available. As an alternative we also propose mappings that do not require a vast extension of the ontology but still provide mappings with similar expressivity. The altenative mapping relations for 'prey' and 'predator' will then be stored as follows:

prey:
sc_domainOf: Animal
[sc_playRole: patient
sc_participantOf: HuntingProcess]
[sc_playRole: patient
sc_participantOf: EatingProcess]

predator:
sc_domainOf: Animal
[sc_playRole: agent
sc_participantOf: HuntingProcess]
[sc_playRole: agent
sc_participantOf: EatingProcess]

B. TMEKO for the perdurant 'bird migration'

The TMEKO procedure for perdurant synsets that are added in the domain wordnets, follows the general steps and interviews generated for non-rigids, with exception for the rigidity interview. The next example shows the interviews and resulting mappings for the synset 'bird migration' to the ontology.

1. Start of the TMEKO procedure
The user adds the synset 'bird migration' to the domain wordnet and relates it to the hypernym 'migration'. The hypernym chain is traversed and returns the perdurant ontology class Migration.

2. Additional confusion interviews (optional)
In the case of perdurant synsets, the user could have been unaware of the distinction between a process and the result of the process as these can be expressed with the same term, as in *air pollution*. As such, the selected hypernym and the according ontology class can be erroneous. A confusion interview for bird migration will not be generated as we don't expect any errors here.

3. Mining definitions
For 'bird migration' the following definitions were found on Google and in the Kyoto document collection:

-*Bird migration is a* seasonal journey that happens twice each year.
-*Bird migration refers to* the regular seasonal journeys undertaken by many species of birds.
-*Bird migration is the* mass intentional and unidirectional movement of a bird population during which time normal stimuli are ignored.

Again, these definitions are processed and the selected words are related to synsets and through the hypernym chain to ontology classes. If the user selects words in the definitions that relate to perdurants like 'journey' or 'movement', no interviews will be generated for these as they are likely to only paraphrase the head of the compound and do not add to useful mapping relations.

4. Presentation and selection of the candidates
The candidate synsets that form the basis for creating mappings to the ontology are presented to the user.
In the case of *bird migration* the following synsets are selected:

-bird:1 (*warm-blooded egg-laying vertebrates characterized by feathers and forelimbs modified as wings*)
Ontology class: Bird
- bird population:1 (*a population of birds*)
Ontology class: Population
-seasonal:1 (*occurring at or dependent on a particular season*)
Ontology class: Regular

5. Create mappings to the ontology
The mapping of *bird migration* to the ontology class Migration is stored as an sc_SubclassOf relation. Each selected synset from the mined definitions is related to an ontology class. Relations of the selected endurants, states and qualities are paraphrased in templates and presented to the user by a set of simple yes/no questions:

-Is it true for [bird migration] that birds participate in this process?
Yes/no → sc_hasParticipant Bird
-Is it true for [bird migration] that the birds themselves are doing something?
Yes/no → sc_hasRole agent
-Is it true for [bird migration] that something/someone does something to/with birds?
Yes/no → sc_hasRole patient

-Is it true for [bird migration] that populations participate in this process?
Yes/no → sc_hasParticipant Population
-Is it true for [bird migration] that the populations themselves are doing something?
Yes/no → sc_hasRole agent
-Is it true for [bird migration] that something/someone does something to/with the population?
Yes/no → sc_hasRole patient

-Is it true for [bird migration] that it happens on a regular basis?
Yes/no → sc_hasState Regular

The positive answers to these questions generate the following mapping relations that are stored in the domain wordnet:

bird migration:
sc_subclassOf: Migration
[sc_hasRole: agent
  sc_hasParticipant: Bird]
[sc_playRole: agent
 sc_hasParticipant: Population]
sc_hasState: Regular

5. Conclusions and future work

TMEKO enables non-expert users to create complex mappings to a shared ontology for both rigid and non-rigid domain synsets. The procedure hides the ontology from the users and translates ontological distinctions to simple natural language questions. The resulting mappings are stored in the domain wordnets and will facilitate crosslingual fact mining and semantic search.
TMEKO is still work in progress; the next phase will involve adding interviews and templates for all classes in the ontology and testing and evaluating of both the system and the quality of the mappings made by the users. As a result, we can evaluate whether or not the co-occurant concepts in the mined definitions generate enough relations that are also useful to the ontology and if the current ontology has the expressivity to facilitate the mappings the users want to make. Secondly, we will investigate how to implement techniques that can check for possible errors and inconsistencies in the mappings, and how long term quality management can be guaranteed without intensive expert supervision. Ultimately, the TMEKO procedure will be translated and adapted for the other six languages involved in the Kyoto project (Dutch, Italian, Spanish, Basque, Japanese and Chinese).

6. Acknowledgments

The KYOTO project is co-funded by EU - FP7 ICT Work Programme 2007 under Challenge 4 - Digital libraries and Content, Objective ICT-2007.4.2 (ICT-2007.4.4): Intelligent Content and Semantics (challenge 4.2). The Asian partners from Taipei and Kyoto are funded from national funds.

7. References

Agirre, E., Lopez de Lacalle, O., Soroa, A. (2009). Knowledge based WSD and specific domains: performing over supervised WSD. In *Proceedings of IJCAI*. Pasadena, USA.

Agirre, E., Soroa, A. (2009). Personalized PageRank for Word Sense Disambiguation. In *Proceedings of the 12th Conference of the European chapter of the Association for Computational Linguistics* (EACL2009). Athens, Greece.

Bosma, W., Vossen, P. (2010). Bootstrapping language neutral term extraction. In *Proceedings of the 7th International Conference on Language Resources and Evaluation* (LREC2010). Malta, May 17-23, 2010.

Bosma, W., Vossen P., Soroa A., Rigau G., Tesconi M., Marchetti A., Monachini M., Aliprandi C. (2009) KAF: a generic semantic annotation format. In *Proceedings of the 5th International Conference on Generative Approaches to the Lexicon* (GL 2009). Pisa, Italy, September 17-19, 2009, pp. 145-152.

Gangemi. A., Guarino, N., Masolo, C., Oltramari, A. (2003) Sweetening Wordnet with DOLCE. In *AI Magazine*. Volume 23, no. 3, pp 13-24.

Guarino, N., Welty C. (2002). Evaluating ontological decisions with OntoClean. In *Communications of the ACM*. Volume 45, no. 2, pp 61-65.

Herold, A., Hicks, A. (2009). Evaluating Ontologies with Rudify Knowledge. In *Proceedings of the International Conference on Knowledge Engineering and Ontology development*. Madeira, Portugal, October 2009.

Ide, N., Romary, L. (2003). Outline of the international standard Linguistic Annotation Framework. In *Proceedings of ACL 2003 Workshop on Linguistic Annotation: Getting the Model Right*, pp 1-5.

Izquierdo, R., Suárez, A., Rigau G. (2007) Exploring the Automatic Selection of Basic Level Concepts. In *Proceedings of the International Conference on Recent Advances on Natural Language Processing* (RANLP'07). Borovetz, Bulgaria, September 2007.

Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A. (2003). *WonderWeb Deliverable D18: Ontology Library*. ISTC-CNR, Trento, Italy.

Oltramari, A., Vetere, G., (2008). Lexicon and Ontology Interplay in Senso Comune. In *Proceedings of OntoLex 2008*. Marrakech, Marocco.

Vossen, P., Agirre, E., Calzolari, N., Fellbaum, C., Hsieh, S., Huang, C., Isahara, H., Kanzaki, K., Marchetti, A., Monachini, M., Neri, F., Raffaelli, R., Rigau, G., Tescon, M., (2008). KYOTO: A system for Mining, Structuring and Distributing Knowledge Across Languages and Cultures. In *Proceedings of the 8th International Conference on Language Resources and Evaluation* (LREC 2008). Marrakech, Marocco, May 28-30, 2008.

Vossen, P. (1998). *Eurowordnet: a multilingual database with lexical semantic networks for European Languages.* Kluwer, Dordrecht.