KAF: a generic semantic annotation format

Wauter Bosma and **Piek Vossen**
CLTL, Vrije Universiteit
Boelelaan 1105, Amsterdam
{w.bosma,p.vossen}@let.vu.nl

Aitor Soroa and **German Rigau**
EHU
San Sebastián
{a.soroa,german.rigau@ehu.es}@ehu.es

Maurizio Tesconi and **Andrea Marchetti**
CNR-IIT
Pisa
{maurizio.tesconi,andrea.marchetti}@iit.cnr.it

Monica Monachini
CNR-ILC
Pisa
monica.monachini@ilc.cnr.it

Carlo Aliprandi
Synthema
Pisa
carlo.aliprandi@synthema.it

Abstract

We present KAF, the KYOTO Annotation Format. KAF is a layered and extendible linguistic annotation format that is specifically developed to arrive at semantic interoperability. KAF is used in seven languages in several applications throughout the KYOTO (Knowledge Yielding Ontologies for Transition-based Organization) project. The goal of these applications is to derive semantic data from linguistically processed text. Separate annotation layers are defined for each annotation process but these can be combined to arrive at a higher level of semantic representation. This paper gives an outline of KAF and a description of how it is applied in the KYOTO project.

1 Introduction

Standardization is essential for interchangibility of data and tools. Once a data format is accepted as a standard, tools can be developed and shared without much data conversion effort. A long-term goal of standardization is to achieve semantic interoperability of content and knowledge. For years, the Semantic Web Community has been working on the standardization of the representation of data (RDF), knowledge (OWL) and services[1] to achieve this. Less progress has been made however with semantic operability of natural language expressions, although this is essential for systems to interact with people that use natural language as their most intuitive interface for communication. The European/Asian KYOTO project[2] aims at establishing semantic interoperability of both knowledge and language to express this knowledge. To achieve this, we anchor words and expressions in language to formal definitions of meaning and use this information to detect knowledge and facts in text. KYOTO tries to establish this across different languages and cultures. Semantic interoperability is achieved by mapping wordnets in each of these languages to a shared ontology, as proposed in the Global Wordnet Grid (Fellbaum and Vossen, 2008), and by means of a common architecture for processing text. The latter is the focus of this paper.

For any of set of languages, KYOTO distinguishes two cycles of text processing:

1. The automatic extraction of terms and concepts, which is performed by *term yielding robots* (Tybots);
2. The automatic detection of facts based on the learned terms and concepts, which is performed by *knowledge yielding robots* (Kybots).

Consequently, the Tybots work in the same way

[1] Semantic Web Services: http://www.w3.org/2002/ws/swsig/

[2] www.kyoto-project.org

for all languages, regardless of their structural properties. To arrive at such semantic interoperability is it necessary to standardize the linguistic processing of text across languages from basic levels of processing such as tokenization up to semantic layers that represent concepts, relations and eventually facts.

There is a range of basic NLP (natural language processing) tasks which are commonly recognized in the field, such as part-of-speech tagging, dependency parsing, etc. As long as every parser produces its output in another proprietary format, their users (i.e., high-level applications, document viewers, etc.) have to deal with a variety of data formats and format conversions. There have been numerous attempts to standardize some aspect of natural language processing. To date, the focus of standards (in various stages of development) includes morphosyntactic annotation (MAF) (Clément and Villemonte de La Clergerie, 2005), syntactic annotation (SynAF) (Declerck, 2006), and semantic annotation (e.g. SemAF[3]).

The beforementioned standards concentrate on a specific stage of annotation. A problem for these formats is that they are difficult to combine. For instance, one might want to do both syntactic annotation and semantic annotation, and integrate the results. The Linguistic Annotation Framework (LAF) (Ide and Romary, 2003) is an ISO standard proposal of a data model for linguistic annotation. It allows individual annotations within the annotation framework to refer to each other, so that the result is a combined analysis of the source text. Rather than a data model, our aim is a layered annotation format, where several processes can add information without losing anything which is produced by a previous process.

In this paper, we present KAF, the KYOTO Annotation Format, or Knowledge Annotation Format. KAF provides annotation layers for basic natural language processing and is open to extensions with other annotation layers needed by specific applications, which may be standardized later on. KAF is compatible with LAF but imposes a more specific standardization of the annotation format itself. In the KYOTO project, we use KAF layers for syntactic annotation such as part-of-speech, compounds, dependency relations and chunks, as well as the semantic layers of semantic role labelling and fact annotation. We show that KAF is adequate for its task by applying it in various applications throughout the KYOTO project across several languages. We also show that KAF can be extended gradually with conceptual layers that can be combined into a presentation of facts expressed in textual documents. For that purpose, we make a distinction between linear annotation in KAF and generic representation of facts that are anchored to the linear annotation as proposed in LAF.

The next section introduces the KYOTO project. Section 3 describes KAF. Section 4 describes two applications in KYOTO which make use of KAF.

2 KYOTO

The globalization of markets and communication brings with it a concomitant globalization of world-wide problems and the need for new solutions. Topical examples are global warming, climate change and other environmental issues related to rapid growth and economic developments. Environmental problems can be acute, requiring immediate support and action, relying on information available elsewhere. Knowledge sharing and transfer are also essential for sustainable growth and development on a longer term. In both cases, it is important that distributed information and experience can be re-used on a global scale. The globalization of problems and their solutions requires that information and communication be supported across a wide range of languages and cultures. Such a system should furthermore allow both experts and laymen to access this information in their own language, without recourse to cultural background knowledge.

The objective of KYOTO is to build a system that allows people in communities to define the meaning of their words and terms in a shared Wiki platform so that it becomes anchored across languages and cultures but also so that a computer can use this knowledge to detect knowledge and facts in text. Whereas the current Wikipedia uses free text to share knowledge, KYOTO represents this knowledge so that a computer can understand it. For example, the notion of environmental *footprint* becomes defined in the same way in all these languages but also in such a way that the computer knows what information is necessary to calculate a *footprint*. With these definitions it becomes possible to find information on footprints in documents, websites and reports so that users can directly ask the computer for information in their environment. KYOTO is a three-year project which started early 2008.

The knowledge cycle in the KYOTO system is

[3]ISO/TC37/SC4 N412, draft

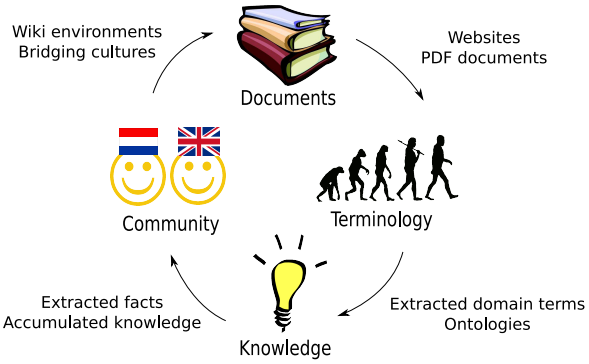

Figure 1: Data flow in the KYOTO system.

outlined in Figure 1. It starts with a set of *documents* produced by the community, such as PDFs and websites. From these documents, the *terminology* is extracted, partly by means of automatic extraction tools (Tybots), and partly by the community by means of editing. This allows users to define "information profiles", specifying information of their interest. For instance, users from the environmental community may be interested in countings of species – data which is present in their set of documents. The Kybots use the information profiles (or Kybot profiles) to extract *knowledge* from documents. *Communities* can affect this process and interact with each other by means of a wiki system which allows them to agree on meaning within a domain and across cultures.

Throughout the KYOTO system, we use text documents at various stages of annotation. Each stage produces a KAF document, adding some information. First, we apply language specific analyses, including tokenization, sentence splitting, part-of-speech tagging, named entity recognition, chunking and dependency parsing. This process is language specific, but the output format is the same KAF format for all languages, so that subsequent processes can be performed in a language neutral manner. Then, we apply word sense disambiguation, requiring the source KAF document and wordnet (Miller et al., 1990) in the document language to produce a new KAF document which includes sense information.

In KYOTO, the resulting KAF document is used by Tybots for automatic terminology extraction, and processed further by Kybots for semantic role labelling. Finally, Kybots aggregate facts which can be presented to the user. We applied terminology extraction to large collections of documents in various languages (so far, English, Dutch, Spanish, Basque). For semantic role labelling and fact aggregation, we developed Kybot proof-of-concept prototypes.

```
<kaf xml:lang="en" doc="example1">
  layer 1...
  layer 2...
  ...
  layer N...
</kaf>
```

Figure 2: An example KAF file, consisting of N layers of annotation.

```
<text>
 <wf wid="w1" page="1" sent="1" para="1">
  Tropical
 </wf>
 <wf wid="w2" page="1" sent="1" para="1">
  terrestrial
 </wf>
 <... skipped ...>
 <wf wid="w16" page="1" sent="1" para="1">
  .
 </wf>
</text>
```

Figure 3: Example: a text layer fragment. Each token (enclosed in a *wf* element) has an identifier, a page number, a sentence number and a paragraph number.

3 KAF: KYOTO Annotation Format

KAF is a layered annotation format, based on XML. If a process adds information which cannot be held by existing layers, a layer of annotation is added. Any previous layers remain intact and can still be used by other processes. Layers may be linked by means of references from one layer to items in another (lower level) layer. Figure 2 shows an example of the general layout of a KAF file.

A full description of the KAF format is given in the KAF manual (Agirre et al., 2009). The remainder of this section gives an overview of the KAF layers of annotation, and relates KAF to ISO standards.

3.1 Syntactic annotation layers

In the KYOTO project, we use KAF in automatic annotation of text documents. In this section, we show annotated examples from different KAF layers for a single sentence:

> *Tropical terrestrial species populations declined by 55 per cent on average from 1970 to 2003.*

KAF provides the following layers to represent the output of common NLP tasks:

```
<terms>
 <term tid="t5" type="open"
    lemma="decline" pos="V">
  <span>
   <target id="w5"/>
  </span>
  <senseAlt>
   <sense sensecode="EN-00441445-v"
     confidence="0.458294"/>
   <sense sensecode="EN-00151689-v"
     confidence="0.541706"/>
  </senseAlt>
 </term>
 <term tid="t7" type="open"
    lemma="per cent" pos="N">
  <span>
   <target id="w8"/>
   <target id="w9"/>
  </span>
 </term>
 <... skipped ...>
</terms>
```

Figure 4: Example: a terms layer fragment. The *span* element contains references to the tokens in the *text layer* which constitute the (multi-)word. The (optional) *senseAlt* element contains references to wordnet senses and their corresponding confidence values.

```
<chunks>
 <!-- tropical terr. species pop. -->
 <chunk cid="c3" head="t4" phrase="NP">
  <span>
   <target id="t1"/>
   <target id="t2"/>
   <target id="t3"/>
   <target id="t4"/>
  </span>
 </chunk>
 <... skipped ...>
</chunks>
```

Figure 5: Example: chunks layer fragment. The *span* element contains references to items in the *terms layer* which constitute the chunk.

```
<deps>
  <!-- tropical, species -->
  <dep from="t1" to="t3" rfunc="mod"/>
  <!-- terrestrial, species -->
  <dep from="t2" to="t3" rfunc="mod"/>
  <!-- species, population -->
  <dep from="t3" to="t4" rfunc="mod"/>
  <!-- population, decline -->
  <dep from="t4" to="t5" rfunc="subj"/>
  <...>
</deps>
```

Figure 6: Example: dependencies layer fragment. For instance, the first *dep* element indicates that *tropical* (the `from` attribute) is a modifier (the `rfunc` attribute) of *species* (the `to` attribute). Both the `from` and the `to` attribute refer to the *terms layer*.

The text layer contains the tokens of the document. Optionally, sentence, paragraph and page boundaries are indicated. This layer – the *text* element in KAF – is the result of sentence splitting and tokenization. Figure 3 shows part of the example sentence, annotated in the text layer.

The terms layer contains words and multi-words. It also includes meta-information such as part-of-speech, references to other resources such as wordnet senses, whether or not it is a named entity, compound elements (in case of a compound), etc. Since (multi-)words consist of tokens, they refer to tokens in the *text layer*. Figure 4 shows examples of (multi-)words in the terms layer.

The chunks layer contains chunks of words, such as noun phrases, prepositional phrases, etc. Since chunks consist of words, they refer to words in the *terms layer*. Each chunk has a *head*, which is also an item in the terms layer. Figure 5 shows an example of a chunk in the chunks layer.

The dependency layer contains dependency relations between words. Since words participate in dependency relations, they refer to words in the *terms layer*. Figure 6 shows examples of dependency relations between words in the example sentence.

The above layers form a chain of dependencies. The base layer of every KAF file is the text layer. All other layers are optional and are founded on the text layer (some indirectly), which makes it compliant with LAF. KAF files with few layers are useful for further processing, or for applications which need only superficial annotation. Although the chunks layer and the dependency layer can be added independently of each other, they are connected by a shared dependency on the terms layer, which ensures that they are both composed of the same elements.

Our objective is a language neutral annotation format. Most of KAF is the same for all languages, but KAF also has facilities for phenomena which are specific for a subset of the languages used in KYOTO. For instance, in order to represent compound nouns explicitly, *term* elements (in the terms layer) which correspond to compounds contain the additional *component* element which includes compound information. Also, information with respect to the declension case can be added.

```
<timexs>
 <!-- 1970 -->
 <timex3 texid="timex1" type="DATE"
    value="1970">
  <span><target id="c7"/></span>
 </timex3>
 <!-- 2003 -->
 <timex3 texid="timex2" type="DATE"
    value="2003">
  <span><target id="c9"/></span>
 </timex3>
 <!-- between 1970 and 2003 -->
 <timex3 texid="timex3" type="DURATION">
    value="P33Y" beginPoint="timex1"
    endPoint="timex2"
    temporalFunction="true"/>
</timexs>
```

Figure 7: Kybot output for a temporal relation, a semantic layer of KAF.

```
<facts>
 <!-- tropical terrestrial species
   declined by around 55 per cent
   between 1970 and 2003 -->
 <fact fid="f1">
  <!-- decline -->
  <process eid="e1"/>
  <!-- around 55 per cent -->
  <quantity qid="q1"/>
  <!-- between 1970 and 2003 -->
  <timex3 texid="timex3"/>
  <!-- tropical terrestrial species -->
  <arg tid="c1" role="patient"/>
 </fact>
</facts>
```

Figure 8: Example: fact annotation fragment.

3.2 Semantic annotation layers

We distinguish two types of annotation: linear annotation and generic annotation. Linear annotation follows the text flow, while generic annotation allows for aggregation of pieces of information throughout the text. The annotation layers in section 3.1 are close to the text, and are linear annotation layers.

In contrast, a generic annotation is a representation of generic knowledge as realized in text. Generic annotation does not necessarily follow the order of the text. Instead, it is centered around knowledge, and how knowledge is anchored in text.

In KYOTO, we plan to generate linear as well as generic annotation layers. SemAF is a linear annotation format which covers annotation of events and expressions of time. Kybots will generate SemAF-compatible annotations as additional layers in KAF. For instance, a Kybot may generate time expressions as specified by SemAF (see Figure 7 for an example). Other Kybots may annotate processes, named entities, co-references, quantities, etc.

Figure 8 shows an example of how we envision generic annotation in KYOTO. The fact annotation layer represents aggregate facts with references to linear annotation layers such as processes, quantities and time.

3.3 KAF and ISO standards for language resources

Given KYOTO's strong vocation towards an open and public system, the KAF data format has been inspired by standard specifications available in the field of Language Resources. MAF and SynAF were investigated as far linguistic annotation for morpho-syntactic and syntactic information, respectively, is concerned. The two meta-models present different degrees of maturity; MAF has entered the last stages of the ISO process, whereas SynAF is at the level of Working Draft standard. The *terms*, *chunks* and *deps* layers of KAF are dedicated to representing morphosyntactic and syntactic information, and are inspired by MAF and SynAF.

Requirements in KYOTO were the representation of syntax, but above all, of semantic annotation. For semantic annotation, the ISO community provides SemAF which is focused on the representation of events and time. We adopted these expressions as separate layers of KAF, making the necessary changes required for integration. An essential difference between the KAF layers and the original SemAF is that SemAF annotation is inserted in the text, while the KAF layers refer to lower level layers (i.e., chunks).

The beforementioned formats focus on a specific type of annotation. In contrast, the goal of KAF is to provide the flexibility of parallel annotations and nevertheless create an integrated view on the document. Applications can use the layers they require. The need for parallel (and possibly independent) annotations also resulted in LAF. KAF is designed to be complementary to LAF: while LAF is a data model for stand-off annotation, KAF can be used to realize LAF in XML structures.

KAF layers are to be seen as dialects of the ISO standards. The KYOTO dialects do not corrupt the compliance with ISO standards and their underlying philosophy. Instead, they are in line with the strategy in ISO which provides high-level models (meta-models) able to be adapted, tailored and implemented according to specific needs.

4 Applications of KAF in the KYOTO project

We describe two applications in KYOTO to exploit KAF annotation: *Tybots* and *Kybots*. The Tybot's job is to automatically extract the domain terminology from a set of KAF-annotated documents. A Tybot produces not just a set of domain-relevant terms, but also relations between them, such as hypernym relations. The domain terms are linked to the wordnet of the corresponding language by means of sense tags in the KAF document which are inserted by our language neutral word sense disambiguation module.

A Kybot detects factual data in the text in various languages. Kybots will be able to detect specific linguistic patterns and semantic relations in text for extracting new facts.

Apart from the Tybots and Kybots, we developed various tools for dealing with KAF files. For instance, we created a KAF viewer which allows the user to analyse a KAF-annotated document. One can view information from KAF layers, such as the structure of sentences and the parts of speech and disambiguated wordnet senses of words. The KYOTO system also includes tools such as a document manager which allows the user to browse document collections, and retrieve original documents.

4.1 Tybots

Rather than starting from scratch, our algorithm for term extraction relies on the annotation layers provided by KAF. This allows us to exploit knowledge of the subjected language while keeping the term extraction algorithm language neutral. We assume that the input is a set of KAF-annotated documents with at least the *chunks layer* (and the layers on which the chunks layer depends). Our general term extraction strategy is the following two-step approach:

1. Extract a large number of candidate terms, and relations between them.
2. Assign a confidence value to each candidate term, representing its domain-relevance – its "termness".

Because a confidence value is associated with each candidate term, an application can set a threshold above which a candidate term is considered a term. A view on the terminology for the selected threshold then shows the terms, a subset of the complete set of candidate terms.

The job of term extraction is performed by a Tybot. Multiple Tybots can be used to build multiple sets of terms. For instance, one Tybot is configured to extract English nouns, while another extracts English adjectives or Dutch nouns. If Tybots produce collections of terms in different languages, they are linked by means of references to wordnets if there is a wordnet mapping for the wordnets in the languages in question. Currently, we apply Tybots to extract nouns from documents in the environmental domain in English, Dutch, Basque and Spanish.

Step 1: candidate terms

The Tybot uses the part-of-speech tags and the chunks to extract candidate terms from the input text. In the case of nouns, the Tybot would extract all nouns and noun phrases. The head of a compound is extracted as another candidate term, which is considered a hypernym of the compound. For instance, the following is derived from the Dutch word *landbouwbeleid* (English: *agricultural policy*):

- Candidate term: *landbouwbeleid*.
- Candidate term: *beleid* (English: *policy*). Hypernym of *landbouwbeleid*.

Extracting chunks allows us to consider more complex candidate terms, such as *terrestrial species*. If *terrestrial species* is encountered as a noun phrase, it is extracted as a candidate term. However, an indication of the concept of *terrestrial species* may be hidden in a longer phrase, e.g. *tropical terrestrial species*. In order to find also these concepts, we extract not only the noun phrase as a candidate term, but also derived candidate terms. Other candidate terms are derived from the noun phrase by transforming the noun phrase (*tropical terrestrial species*) into its head (*species*) by removing the non-head words one by one from the head or the tail of the noun phrase. The resulting phrase after each removal operation is considered a candidate term with a hypernym relation to all other candidate terms derived from this intermediate phrase. In the example of *tropical terrestrial species*, the following is derived from this phrase:

- Candidate term: *tropical terrestrial species*.
- Candidate term: *terrestrial species* (removed: *some*). Hypernym of *tropical terrestrial species*.
- Candidate term: *species* (removed: *terrestrial*). Hypernym of *terrestrial species*.

Step 2: domain relevance

Each candidate term is assigned a confidence value, representing its relevance to the domain. Candidate terms above a certain threshold are considered *terms*. Most thesaurus extraction algorithms are frequency-based: a more frequent word is more likely to be a domain-relevant term than a less frequent word. In our application, just counting the occurrence frequency of candidate terms is flawed because the same document may be offered as input twice, documents may overlap and documents may repeat a short piece of text (such as the title of the document or a line like *call us for more information*).

To determine domain-relevance, we decided not to use just occurrence frequencies, but also the position of a candidate term in the hierarchy – the hypernym relations between candidate terms. The confidence value of a candidate term is a value between 0 and 1, derived from the number of direct and indirect hyponyms of the candidate term, and the sum of their document frequencies (the document frequency of a term is the number of documents in which the term occurs) of all hyponyms.

4.2 Kybots

Once the ontological anchoring is established by the Tybots, it is be possible to build text mining software that can detect semantic relations and facts occurring among concepts already integrated into the ontologies. Thus, the Kybots will produce enriched KAF outputs, incorporating new layers of semantic knowledge or facts.

There are two types of Kybots. Kybots of the first type, level-1 Kybots, perform semantic analysis over KAF documents and create new linear layers on top of the existing layers, as described in section 3.2. That is, enriching KAF with new information structures. Layer-1 Kybots deal with processes like named entity recognition, coreference resolution, quantity identification, annotation of time expressions, etc.

On the other hand, level-2 Kybots will extract facts by analyzing the semantic information level-1 Kybots produce on KAF document collections. Therefore, facts can be extracted by aggregating information from different linguistic information layers, documents or even different languages. The facts extracted by level-2 Kybots will be represented in generic annotation layers. Fox example, Figure 8 shows a fact which is extracted by combining information from two sentences. Note that this Kybot relies on a co-reference annotation layer.

The level-2 Kybots are text miners which will be defined by linguistic patterns and semantic constraints expressed at an ontological level. For example, the ontology will give us the conceptual pattern that Populations consist of species that live in a habitat in some region. This information can be realized through e.g. compounding as in: Mediterranean spider population, or as a sentence as in: Large groups of alien spiders that live in dry areas in Mediterranean mountain areas.

In fact, the Kybots will provide a mapping between the conceptual constraints and the linguistic patterns.

The facts of interest are defined in so-called Kybot profiles. The profiles can be defined in advance or by individual users. An initial design has been set-up allowing to characterize Kybot profiles in terms of:

Expression Rules: conditions on the Linguistic Processing outcomes, flexible enough for dealing with all KAF outputs and to capture some information from KAF.

Semantic Conditions: ontological conditions the captured information must satisfy.

Output Template: extracted output expressions, consistent with the ontology.

Figure 9 shows an example of a Kybot profile for locating expressions involving a decrease predicate followed by a percentage. The Kybot profile has three main parts:

- Declaration of variables (X, Y and Z in the example).
- Declarations of the relations among variables. Typical relations are *following*, *preceding*, *window*, etc. If the relations among these variables hold, a matching is produced.
- The output format refering to variables previously defined.

Once the Kybot profile has been defined, the system will check and compile it. The resulting Kybot can be applied to the analysed text (a KAF file). Thus, for each analysed sentence a Kybot will be applied following:

IF *Expression Rules* match and *Semantic Conditions* hold

THEN generate the Output Template.

```
<?xml version="1.0" encoding="utf-8"?>
<Kybot id="decrease-by-Z\%">
 <variables>
  <var name="X" value="term[starts-with(@pos,'v') and .//sense[@sensecode='00111597-v']]"/>
  <var name="Y" value="term[starts-with(@pos,'p')]"/>
  <var name="Z" value="term[ends-with(@lemma,'%') or ends-with(@lemma,'percent')]"/>
 </variables>
 <relations>
  <root span="X"/>
  <rel span="Y" pivot="X" direction="following" dist="1"/>
  <rel span="Z" pivot="Y" direction="following" dist="1"/>
 </relations>
 <facts>
  <fact id="quantity-change-001">
   <factval name="term" value="$Z/@tid"/>
   <factval name="quantity" value="$Z/@lemma"/>
  </fact>
 </facts>
</Kybot>
```

Figure 9: Example of a Kybot profile.

The KYOTO system includes a Kybot editor, which is used to define Kybot profiles, using expression rules for the *terms layer* of KAF. The Kybot editor allows the user to (a) upload and manage annotated documents (in particular KAF documents), (b) search words or terms in the collection of uploaded documents, and (c) create and execute Kybots based on specific users information needs.

We also designed an initial scenario for the Kybot profile construction we called *mining by example*. In this scenario, an interface is provided to the user, allowing for the construction of Kybots from corpus examples, without the need to access the complex conceptual patterns and the linguistic structures. Collections of Kybots created this way will be applied to extract relevant knowledge from textual sources in different languages and cultures.

5 Conclusions

KAF is a layered annotation format which provides layers for commonly used linguistic structures as well as semantic information. KAF can be extended with layers for specific applications. KAF provides a mechanism for referring from one layer to items in another layer, so that an integrated view on a document can be constructed by gradually adding layers of annotation. In KYOTO, we use KAF in several applications for processing text in seven languages.

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 Tapei and Kyoto are funded from national funds.

References

Eneko Agirre, Xabier Artola, Arantza Diaz de Ilarraza, German Rigau, Aitor Soroa, and Wauter Bosma. 2009. KAF: Kyoto Annotation Framework. Technical Report TR 1-2009, Dept. Computer Science and Artificial Intelligence, University of the Basque Country.

Lionel Clément and Éric Villemonte de La Clergerie. 2005. MAF: A morphosyntactic annotation framework. In *Proceedings of the 2nd Language & Technology Conference*, pages 90–94.

Thierry Declerck. 2006. SynAF: Towards a standard for syntactic annotation. In *Proceedings of the Fifth Conference on International Language Resources and Evaluation*, pages 229–233. European Language Resources Association (ELRA).

Christiane Fellbaum and Piek Vossen. 2008. Challenges for a Global WordNet. In J. Webster, Nancy Ide, and A. Chengyu Fang, editors, *Online Proceedings of the First International Workshop on Global Interoperability for Language Resources*, pages 75–82, January.

Nancy Ide and Laurent Romary. 2003. Outline of the international standard Linguistic Annotation Framework. In *Proceedings of ACL 2003 Workshop on Linguistic Annotation: Getting the Model Right*, pages 1–5. Association for Computational Linguistics.

George Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. 1990. Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–244.