

---

# Unsupervised Ontology-based Semantic Tagging for Knowledge Markup

---

Paul Buitelaar  
Srikanth Ramaka

PAULB@DFKI.DE  
SRIKANTH.RAMAKA@DFKI.DE

DFKI GmbH, Language Technology, Stuhlsatzenhausweg 3, 66123 Saarbruecken, Germany

## Abstract

A promising approach to automating knowledge markup for the Semantic Web is the application of information extraction technology, which may be used to instantiate classes and their attributes directly from textual data. An important prerequisite for information extraction is the identification and classification of linguistic entities (single words, complex terms, names, etc.) according to concepts in a given ontology. Classification can be handled by standard machine learning approaches, in which concept classifiers are generated by the collection of context models from a training set. Here we describe an unsupervised approach to concept tagging for ontology-based knowledge markup. We discuss the architecture of this system, and our strategy for and results of performance evaluation.

## 1. Introduction

A central aspect of Semantic Web development is knowledge markup: annotation of data with formalized semantic metadata in order to allow for automatic processing of such data by autonomous systems such as intelligent agents or semantic web services (see e.g. McIlraith et al., 2001). As much of today's information is available as text only, knowledge markup often involves the annotation of textual data to explicitly structure the knowledge that is available in text only implicitly. Automating this process involves the use of information extraction technology that allows for the mapping of linguistic entities (single words, complex terms, names, etc.) to shallow semantic representations, mostly referred to as 'templates' (see e.g. Ciravegna, 2003). Consider for instance the following example from the football domain, which expresses a typical event with a number of roles to be filled by information extraction from relevant textual

data, e.g.: *In the last minute Johnson saved with his legs from Huckerby*

```
RESCUE-EVENT [  
  goalkeeper : GOALKEEPER > Johnson  
  player      : PLAYER      > Huckerby  
  manner      : BODYPART    > legs  
  atMinute    : INT          ] > 90
```

Obviously, if such templates are expressed in a formally defined knowledge markup language such as RDFS or OWL, they roughly correspond to an ontologically defined class with its attributes (properties). In the context of this paper we therefore assume an interpretation of information extraction for knowledge markup as *concept instantiation*<sup>1</sup> that includes:

- concept tagging – mapping of linguistic entities to concepts/classes as defined by an ontology
- attribute filling – mapping of linguistic structure over linguistic entities that are tagged with a class to attributes of that class as defined by an ontology

Here we focus primarily on concept tagging, which is a prerequisite for attribute filling. We treat concept tagging as a classification task that can be handled by standard machine learning approaches, in which concept classifiers are generated by the collection of context models from a training set. Context models may be generated from manually annotated, i.e. *supervised* training sets, but this is very costly and non-robust as for each new ontology a supervised training set needs to be constructed. Instead, we present development of an *unsupervised* approach that can be trained on any relevant training data, without previous manual annotation.

---

Appearing in *W4: Learning in Web Search at 22<sup>nd</sup> International Conference on Machine Learning*, Bonn, Germany, 2005. Copyright 2005 by the author(s)/owner(s).

---

<sup>1</sup> Concept instantiation has also been referred to as 'ontology population' (e.g. in the context of the AKT project - <http://www.aktors.org/akt/>), which emphasizes the database aspect of an ontology and its corresponding knowledge base.

This is similar to the SemTag approach to large-scale semantic tagging for the Semantic Web (Dill et al., 2003), but the emphasis of our approach is somewhat different. We focus here on an unsupervised approach to concept tagging as a necessary prerequisite for further information extraction and more complex knowledge markup, whereas the SemTag approach emphasizes the large-scale aspects of concept tagging without a clear vision on the eventual use of the added semantic tags.

The remainder of the paper gives an overview of the system architecture of our approach in section 2, followed in section 3 by a discussion of our evaluation strategy and results of this. In section 4 we give an outline of the application of the system in two Semantic Web projects. Related work is presented in section 5.

## 2. System Architecture

The unsupervised concept tagging system we are developing consists of the following components:

- a set of hierarchically organized classes from a domain ontology
- a domain-relevant document collection for training and classification
- a shallow linguistic module for preprocessing class labels and documents
- a machine learning environment for generating context models and classifiers
- a knowledge base to store marked up concept instantiations

In the training phase, a context model and classifier is generated from a domain-specific document collection for a set of classes from a corresponding domain ontology, over which various parameters are evaluated to select the best classifier. In the application phase, the classifier is used in tagging linguistic entities with the appropriate class and to store corresponding class instances in the knowledge base. In information extraction, these instances (with linguistic contexts) are submitted to a further process that maps them to relevant class attributes. We will not address this any further here, but applications of the information extraction process are discussed in section 4.

### 2.1 Ontology and Document Collection

The system assumes as primary input an ontology in RDFS or OWL with a hierarchy of classes as specified for a particular domain. The following two example classes from the ‘‘Soccer V2.0’’ ontology<sup>2</sup> on football express two

<sup>2</sup> Available from <http://www.lgi2p.ema.fr/~ranwezs/ontologies/soccerV2.0.daml>, which we adapted to OWL and added German labels.

events (‘to clear’ and ‘counter attack’) that are defined as sub-classes of a class that expresses the more general event ‘other player action’<sup>3</sup>:

```
<rdfs:Class rdf:ID="Clear">
  <rdfs:subClassOf
    rdf:resource="#Other_player_action"/>
  <rdfs:label
    xml:lang="en">Clear
  </rdfs:label>
  <rdfs:label
    xml:lang="de">Klären
  </rdfs:label>
</rdfs:Class>

<rdfs:Class rdf:ID="Counter_attack">
  <rdfs:subClassOf
    rdf:resource="#Other_player_action"/>
  <rdfs:label
    xml:lang="en">Counter_attack
  </rdfs:label>
  <rdfs:label
    xml:lang="de">Konterangriff
  </rdfs:label>
</rdfs:Class>
```

Next to a domain ontology, the system assumes a document collection on the same domain. For instance, for the SmartWeb project<sup>4</sup> that will be discussed in Section 4 below, we are working with a football ontology and a document collection on UK football matches<sup>5</sup>.

### 2.2 Linguistic Preprocessing

In order to map linguistic entities in the document collection on classes in the ontology, we normalize them into a common linguistic representation. For this purpose we linguistically preprocess the class names in the ontology as well as all text segments in the document collection.

Linguistic preprocessing<sup>6</sup> includes part-of-speech (PoS) tagging with the TnT tagger (Brants, 2000) and lemmatization based on Mmorph (Petitpierre and Russell, 1995). Part-of-speech tagging assigns the correct syntactic class (e.g. noun, verb) to a particular word given its context. For instance, the word *works* will be either a verb (*working the whole day*) or a noun (*all his works have been sold*).

<sup>3</sup> We use the OWL API (Bechhofer et al., 2003) in parsing the ontology.

<sup>4</sup> More information on the SmartWeb project can be obtained from <http://www.smartweb-projekt.de>

<sup>5</sup> The football document collection used here is obtained by crawling a web portal on premiere league football in the UK: <http://4thegame.com>

<sup>6</sup> Linguistic preprocessing is accessed via an XML-based format based on proposals in (Buitelaar and Declerck, 2003).

Lemmatization involves normalization over inflectional, derivational and compound information of a word. Inflectional information reduces the plural noun *works* to the lemma *work*, whereas derivational information reduces the verb forms *working* and *works* to the lemma *work*. Compound information determines the internal structure of a word. In many languages other than English the morphological system is very rich and enables the construction of semantically complex compound words. For instance the German word “*Schiedsrichterfahne*” corresponds in English with two words “*referee flag*”.

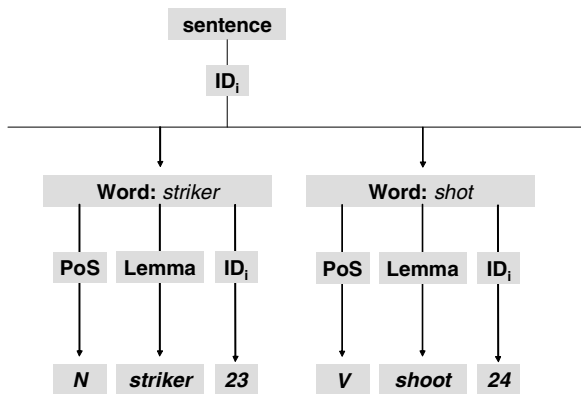


Figure 1: Linguistic Annotation Example

### 2.3 Generating Context Models and Classifiers

The concept tagging system is based on an instance-based learning approach to classification as implemented for instance in the WEKA machine learning environment. Instance-based learning involves a nearest neighbor classification method, in which the instance to be classified  $i$  is compared with all training instances, using a distance metric, and the closest training instance is then used to assign its class to  $i$ . The generalization of this method that we use here is the  $k$ -nearest neighbor method, where the class of the instance  $i$  is computed using the closest  $k$  training instances.

An instance-based learning algorithm consists of a training step and an application step. We first discuss the training step, in which context models and corresponding classifiers are generated. In the next sub-section we discuss the application of such classifiers in concept tagging.

Training involves the construction of classified instances from a training set. As the methods discussed here are unsupervised, this training set has not been previously annotated. An instance is a set of attribute-value pairs, one of which identifies the class that needs to be determined.

Constructing an instance involves the following. Let  $w$  be a word in the training set, for which we can build

instances with the attribute-value pairs of each instance filled by its left and right neighbor words in a context of size  $N$ . The attribute-value pair that represents the class of this instance is filled by matching the word  $w$  with the preprocessed class name and the class names of all of its sub-classes. To illustrate the construction of particular instances, consider the following sentences from the document collection on football:

*Even during those early minutes Palace's former Carlisle **attacker** Matt Jansen looked up for a big game, and no wonder as he was facing his boyhood idols!*

*Arsenal's new French **midfielder** Patrick Vieira started the rot for Leeds this time after only 44 seconds.*

*That they went home empty-handed was largely down to another of Gullit's instant imported hits, former Strassbourg **sweeper** Frank Leboeuf.*

The words *attacker*, *midfielder*, *sweeper* match with the classes **attacker**, **midfielder**, **sweeper** in the football ontology, which are sub-classes of the class **player**. From the sentences we may now derive the following instances for this class with context size 5 (2 words on the left, 2 words on the right):

N-2	N-1	N+1	N+2
former	Carlisle	Matt	Jansen
new	French	Patrick	Vieira
former	Strassbourg	Frank	Leboeuf

In this way, we can build up a context model and corresponding classifier for each class. In the application phase these classifiers will be used to classify unseen terms. Consider for instance the word *striker* in the following sentence:

*The big French **striker** stepped up to drill home the penalty himself.*

The word *striker* (in this context) expresses the sub-class **striker** of the class **player**, which has not been modeled as such in the football ontology. We therefore can use classification to extend the coverage of the concept tagging system and at the same time to acquire additional sub-classes for each of the classes modeled in the training step. In this way, knowledge markup can be connected to ontology learning, which aims at automatic or semi-automatic extension and/or adaptation of ontologies<sup>7</sup>.

<sup>7</sup> See the collection of papers from the ECAI04 workshop on Ontology Learning and Population for an overview of recent work <http://olp.dfki.de/ecai04/cfp.htm>.

## 2.4 Classification: Concept Tagging

In the application step, we use the generated classifiers to classify an occurrence of word  $w$  by finding the  $k$  most similar training instances. For instance, for the sentence with *striker* above, we extract the corresponding instance to be classified (with the class missing):

```
[big, French, stepped, up, -]
```

Now we classify the instance using the generated classifiers to obtain:

```
[big, French, stepped, up, player]
```

The output of this process is a classified instance that will be represented in two ways:

- Concept Tagging – mark up of corresponding tokens in the document with the assigned class in XML<sup>8</sup>
- Knowledge Base Instantiation – generation of an RDF instance for the assigned class in the ontology (with a pointer to corresponding tokens in the document)

To illustrate this, consider the example in Figure 2 below. Here, the word *striker* is marked as **player** with an indication of the origin of this class through the information stored in the `ontology` attribute. An instance in RDF can be created accordingly and stored in the knowledge base.

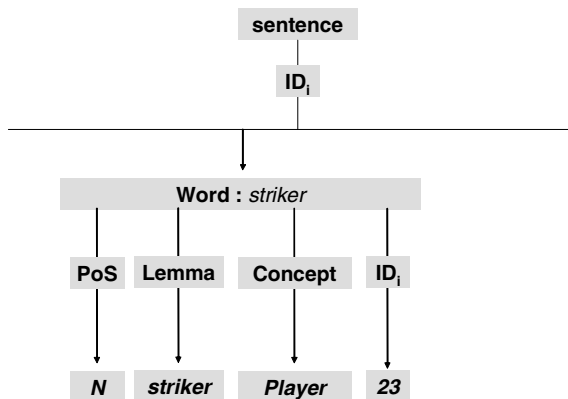


Figure 2: Concept Tagging Example

## 3. Evaluation

An important step in system development is performance evaluation, in order to determine the appropriate research direction for the task at hand. In the context of this paper

<sup>8</sup> Concept tagging extends the XML output of linguistic preprocessing as discussed in section 2.2 (see also Buitelaar and Declerck, 2003)

we were interested to determine an answer to the following research questions:

1. How well does the system perform on correctly classifying new terms (i.e. terms that are not yet modeled in the ontology)?
2. What is the influence of linguistic preprocessing (PoS tagging, lemmatization) on classification results?

In this section we discuss our strategy in evaluating these questions, the evaluation set we constructed and the results obtained with this evaluation set.

### 3.1 Evaluation Strategy

To evaluate our approach we developed a performance evaluation strategy that assumes a gold standard with which different data sets can be automatically compared and on the basis of which recall and precision numbers can be computed in a straightforward way. A major criticism of performance evaluation is that it evaluates only the clearly measurable aspects of the technology used, without taking the wider user-oriented context into account. Although this is surely correct from a wider user-oriented perspective, for comparing results on many different parameters there seems to be no alternative to the use of a gold standard. We therefore developed a gold standard classification set for the football domain, derived from the document collection and football ontology mentioned earlier.

### 3.2 Evaluation Sets

The gold standard was constructed by pooling: running the system with the same data set over a number of different parameters (context size, value of  $k$ ). We then merged the resulting classified data sets by taking the intersection of all classified instances. This resulted in an evaluation set of 869 classified instances that we gave to three evaluators to judge on correctness<sup>9</sup>. The task of the evaluators was to judge if a word  $w$  was correctly classified with class  $c$ , given its context (sentence)  $s$ . The classified instances were presented to the evaluator as follows:

$c$ : **other\_player\_action**

$w$ : *volleying*

$s$ : *Wiltord fed the ball through to Dennis Bergkamp and his chip into Henry's path led to the French striker volleying over from six yards when it appeared easier to score.*

The evaluators were then asked to judge this classification by assigning it a 3 (very good), 2 (good), or 1 (incorrect). We were able to assemble a gold standard from these

<sup>9</sup> The evaluators qualified as `domain experts` as they were all football aficionados.

judgments by taking a voting account of the three assignments for each classified instance. For 863 instances a majority could be established in this way, for the remaining 6 instances each evaluator assigned a different score. These instances were therefore left out of the resulting gold standard.

The 863 instances in the gold standard are distributed over 4 classes in the football ontology that we selected for evaluation:

**other\_player\_action** with sub-classes: **beat, charge, clear, ...**

**person** with sub-classes: **official, player, ...**

**place** with sub-classes: **area, field, line, ...**

**stoppage** with sub-classes: **corner, fault, goal, ...**

The distribution of judgments over these classes is as follows:

Table 1: Distribution of judgments over the 4 selected classes

	<i>very good</i>	<i>good</i>	<i>incorrect</i>
<b>other_player_action</b>	47	32	104
<b>person</b>	50	4	57
<b>place</b>	24	14	118
<b>stoppage</b>	4	2	407
<b>Total</b>	125	52	686

From the set of evaluated instances we then created two gold standard evaluation sets, a “strict” one (including only the instances judged to be classified “very good”) and a “relaxed” one (including the “very good” as well as the “good” instances). The “strict” set has 125 and the “relaxed” set 177 instances.

### 3.3 Evaluation Results

We used the two gold standard sets to evaluate different settings for  $N$  (context size) and the number of closest  $k$  training instances. To evaluate the influence of context size we varied  $N$  between 1, 2 and 5, each time with  $k$  between 1, 2 and 10. The results are presented in the following tables.

The results in table 2 show that a larger context size degrades recall significantly as we consider only contexts within sentence boundaries. Obviously, there are more n-

grams of length 3 ( $N=1$ ) than of length 11 ( $N=5$ ) within a sentence. The influence of  $k$  seems not significant, although  $k=1$  gives the best results at  $N=1$ .

Table 2: Evaluation results

$N$	$k$	<i>Strict Set</i>		<i>Relaxed Set</i>	
		<i>Rec.</i>	<i>Prec.</i>	<i>Rec.</i>	<i>Prec.</i>
1	1	89% (111)	89% (99)	92% (162)	89% (144)
	2	89% (111)	87% (97)	92% (162)	87% (141)
	10	87% (109)	83% (90)	89% (158)	84% (132)
2	1	69% (86)	83% (71)	66% (117)	82% (96)
	2	66% (82)	85% (70)	64% (114)	82% (94)
	10	66% (83)	84% (70)	65% (115)	82% (94)
5	1	17% (21)	81% (17)	18% (31)	81% (25)
	2	15% (19)	84% (16)	16% (29)	76% (22)
	10	14% (18)	83% (15)	15% (27)	81% (22)

The results in table 2 provide an answer to our first research question (how well do we classify?). The answer to the second question (does linguistic preprocessing matter?) is given by the results in the following table. In this case we did not use any linguistic preprocessing in training and application. As the table shows, the results are worse than with linguistic preprocessing (only results for  $N=1$  are shown).

Table 3: Evaluation results – no linguistic preprocessing

$N$	$k$	<i>Strict Set</i>		<i>Relaxed Set</i>	
		<i>Rec.</i>	<i>Prec.</i>	<i>Rec.</i>	<i>Prec.</i>
1	1	74% (92)	85% (78)	76% (135)	84% (113)
	2	74% (92)	83% (76)	76% (135)	81% (110)
	10	74% (92)	79% (73)	75% (132)	79% (104)

## 4. Application

The concept tagging system described in this paper is being developed in the context of two projects (SmartWeb, VieWs) that we are currently working on. The projects have different scenarios and application domains, but share a need for tagging of text documents with classes from a given ontology for information extraction purposes.

### 4.1 SmartWeb

SmartWeb is a large German funded project that aims at intelligent, broadband mobile access to the Semantic Web. For this purpose it combines such diverse technologies as speech recognition, dialogue processing, question answering, information extraction, knowledge management and semantic web services into an ambitious common framework to realize an intelligent mobile information system.

A first demonstrator is targeted to the football world cup 2006, which will be held in Germany. The SmartWeb system will be able to assist the football fan over speech input in booking his tickets for the games he wants to see, as well as hotels, restaurants, etc. Additionally, the system will be able to answer questions on any football related issue (e.g. game history, end scores, names and achievements of players) or otherwise (e.g. the weather, local events, news).

In order to be able to answer such questions, the system will need to have knowledge of many topics which will be handled by a combination of several technologies: open-domain question answering on the web (based on an information retrieval approach), semantic web service access to web-based databases and ontology-based information extraction from football related web documents for knowledge base generation. Concept tagging with the SmartWeb football ontology is a prerequisite for the ontology-based information extraction task.

### 4.2 VieWs

The VieWs<sup>10</sup> project has as its central aim to demonstrate how web portals can be dynamically tailored to special interest groups. The VieWs system combines ontologies, information extraction, and automatic hyperlinking to enrich web documents with additional relevant background information, relative to particular ontologies that are selected by individual users. A tourist for instance will be shown additional information on hotels, restaurants or cultural events by selecting the tourist ontology.

On entering a VieWs enhanced web portal the system analyses the web document provided by the server and

identifies anchors for the hyperlinks, e.g. city names. A Google-based web search is then started for the recognized city names in combination with keywords (“hotel”, “restaurant”, etc.) derived from the ontology.

The results of the web search and information already existing in the knowledge base will be shown in the form of generated hyperlink menus on each of the identified city names. Additionally, an information extraction process is started in the background over the retrieved documents and relevant extracted information is stored in the knowledge base for future access. Obviously also here ontology-based concept tagging is a prerequisite for the information extraction process.

## 5. Related Work

As mentioned before, the work discussed here is related to the SemTag work on large-scale semantic tagging for the Semantic Web (Dill et al., 2003). Also much of the work on semantic annotation (for a recent overview see: Handschuh and Staab, 2003) and ontology learning (for a recent overview see: Buitelaar et al., 2005) for the Semantic Web is directly related. However, next to this also various other tasks in natural language processing and information retrieval are concerned with similar issues.

First of all, the large body of work on semantic tagging and word sense disambiguation is of direct interest as this is also concerned with the assignment of semantic classes to words (for an overview see Ide and Veronis, 1998; Kilgarriff and Palmer, 1999; Edmonds and Kilgarriff, 2003). However, there is also an important difference as this work has been almost exclusively concerned with the use of lexical resources such as dictionaries or wordnets for the assignment of semantics to words in text. The use of ontologies brings in a rather different perspective, e.g. on lexical ambiguity, on lexical inference and on the mapping of linguistic structure to semantic structure.

A second important area of related work is named-entity recognition (for a recent overview see e.g. Tjong Kim Sang and De Meulder, 2003). Named-entity recognition (NER) is also concerned with the assignment of semantic classes to words or rather names in text. However, the typical number of semantic classes used in NER is mostly small, not extending beyond distinctions such as person, location, organization, and time. Nevertheless, there is an important overlap in the methods and goals of NER and the work discussed here, that is if we imagine NER with a larger and hierarchically ordered set of semantic classes as specified by an ontology. Such a direction in NER has been given much consideration lately, as witnessed for instance by the SEER<sup>11</sup> (Stanford Edinburgh Entity Recognition) project.

---

<sup>10</sup> <http://views.dfki.de>

---

<sup>11</sup> <http://www.ltg.ed.ac.uk/seer/>

## 6. Conclusions

We presented ongoing work on developing an ontology-based concept tagging system as an important prerequisite in information extraction for knowledge markup. The system we discussed implements an unsupervised approach, in which no prior manual tagging is needed. Such an approach allows for a robust application of the system in different domains. Evaluation indicates that good results can be obtained with such an approach and that linguistic preprocessing helps to increase recall and precision.

## Acknowledgements

This research has been supported by grants for the projects VIEWS (by the Saarland Ministry of Economic Affairs) and SmartWeb (by the German Ministry of Education and Research: 01 IMD01 A).

## References

- Bechhofer Sean, Phillip Lord, Raphael Volz. *Cooking the Semantic Web with the OWL API*. 2nd International Semantic Web Conference, ISWC, Sanibel Island, Florida, October 2003.
- Brants, Thorsten. *TnT - A Statistical Part-of-Speech Tagger*. In: Proceedings of 6th ANLP Conference, Seattle, 2000.
- Buitelaar, Paul and Thierry Declerck. *Linguistic Annotation for the Semantic Web*. In: Handschuh S., Staab S. (eds.) Annotation for the Semantic Web, IOS Press, 2003.
- Buitelaar, Paul, Philipp Cimiano and Bernardo Magnini (eds.) *Ontology Learning from Text: Methods, Evaluation and Applications*. IOS Press, 2005.
- Ciravegna, Fabio. *Designing adaptive information extraction for the semantic web in amilcare*. In Siegfried Handschuh and Steffen Staab, editors, Annotation for the Semantic Web, Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam, 2003.
- Dill S., N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, T. Kanungo, S. Rajagopalan, A. Tomkins, J. A. Tomlin, and J. Y. Zien, *SemTag and Seeker: Bootstrapping the semantic Web via automated semantic annotation*, 12<sup>th</sup> International World Wide Web Conference Budapest, Hungary, 2003.
- Edmonds, Phil and Adam Kilgarriff (eds.). *Journal of Natural Language Engineering (special issue based on Senseval-2)*, vol.9 no. 1, Jan. 2003.
- Handschuh, Siegfried and Steffen Staab (eds.) *Annotation for the Semantic Web*. IOS Press, 2003.
- Ide, N. and Veronis J. *Introduction to the special issue on word sense disambiguation: The state of the art*. *Computational Linguistics*, 24(1):1--40. 1998.
- Kilgarriff, Adam and Martha Palmer (eds.). *Computers and the Humanities (special issue based on Senseval-1)*, vol.34 no. 1-2, 1999.
- McIlraith, Sheila A., Tran Cao Son, and Honglei Zeng *Semantic Web Services* IEEE Intelligent Systems, March/April 2001, Vol 16, No 2, pp. 46-53.
- Petitpierre, D. and Russell, G. *MMORPH - The Multext Morphology Program*. Multext deliverable report for the task 2.3.1, ISSCO, University of Geneva. 1995.
- Tjong Kim Sang, Erik F. and Fien De Meulder. 2003. *Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition*. In: Walter Daelemans and Miles Osborne (eds.), *Proceedings of CoNLL-2003*, Edmonton, Canada.