OCAS: Ontology-Based Corpus and Annotation Scheme

Towards an OBIE Gold Standard that contains even implicit facts

Alexander Grothkast[1,2], Benjamin Adrian[1], Kinga Schumacher[1], and Andreas Dengel[1,2]

[1] Knowledge Management Department, DFKI, Kaiserslautern, Germany
[2] University of Kaiserslautern, Kaiserslautern, Germany
firstname.lastname@dfki.de

Abstract. This paper presents strategies and lessons learned from the creation of a corpus. It suggests a gold standard for evaluating ontology-based information extraction (OBIE) systems. This OBIE gold standard is called OCAS2008 and consists of: (i) an OBIE layer cake for comparing OBIE systems by subtasks, (ii) a document corpus of 121 documents with 31,000 words about a closed domain, (iii) a compact domain ontology including more than 40,000 instances, (iv) two annotation scenarios that extend traditional template-based evaluations, (v) an annotation set that contains typed annotations according to the ontology and the OBIE layer cake, (vi) annotations that concern text phrases, symbols, instances, explicitly written facts, implicit facts, and (vii) finally, human created annotations according to predefined specifications. We claim that the use of OCAS2008 provides a basis for comparable and significant evaluations of OBIE systems.

1 Introduction

Information Extraction (IE), as introduced in the *Message Understanding Conference* (MUC[3]) series and proceeded in succeeding *Automated Content Extraction* (ACE[4]) competitions, is known for significant evaluations of its IE tasks. Traditional IE systems are evaluated in units of IE subtasks. IE subtasks were firstly described in *Hobbs' Generic IE system* [1] that was developed during the Tipster program. It forms the base of modern IE systems. Emerging ontology-based IE approaches (e.g., [2], [3], [4], [5]) claim to enhance traditional IE by supporting domain adaptability, and to extract even implicit information by using inference mechanisms. Apart from these benefits, an analysis of current OBIE approaches reveals weaknesses in evaluating and comparing these [6]. One reason is, that OBIE approaches enhance IE functionality, but do not agree in a Generic OBIE system as done in traditional IE systems. This results in heterogeneous architectures that are hard to compare. Evaluation costs increase even more as traditional IE evaluation methods do not suffice [6]. Therefore, this work describes methodologies called *Ontology-Based Corpus and Annotation Scheme* for creating the OBIE gold standard OCAS2008. OCAS2008 consists of:

[3] http://www.itl.nist.gov/iaui/894.02/related_projects/muc
[4] http://www.nist.gov/speech/tests/ace/

– a generic OBIE architecture called *OBIE layer cake* for comparing OBIE systems by similar subtasks,
– a document corpus of 121 news articles with 31,000 words about a closed domain (Olympic Summer Games 2004),
– a compact domain ontology about the Olympic Summer Games 2004 including more than 40,000 instances,
– two annotation scenarios that extend traditional template-based evaluations,
– an annotation set that contains typed annotations according to the ontology and the *OBIE layer cake*,
– annotations that concern text segments, symbols, instances, explicitly written facts, implicit facts, and
– finally, human created annotations according to predefined specifications.

The paper is structured as follows. We begin describing the current efforts in evaluating OBIE systems and how gold standards are built. Concluding this, we discuss the creation of our text corpus about the Olympic Summer Games 2004 called OCAS2008 along a four-step process in Section 3. Finally, the paper summarizes the proposed OCAS2008 and gives an outlook which OBIE systems will be evaluated against this gold standard.

2 Related Work

Within MUC-5 in 1993 and the Tipster program, HOBBS [1] introduced a generic view on traditional IE systems and their subtasks. One famous IE system is the General Architecture for Text Engineering (GATE) [7]. Such systems have been evaluated in the MUC series [8] and ACE competitions [9] using text corpora and templates. The Linguistic Data Consortium [10] describes various aspects and best practices of corpus creation. These systems are evaluated based on discrete metrics such as *precision*, *recall*, *f-measure*, or *MUC error measure* [11]. Apart from traditional IE, in OBIE ontologies are used for representing domain knowledge as done in GATE [3]. The SEKT project used GATE as OBIE system and created the annotated *OntoNews*[12]. This corpus consists of 292 news documents on UK politics, international politics, and business. We account these huge domains as too wide spread and not completed for using them in a significant evaluation. Another example for a large annotated corpus is the ACE2004 training set. It captures nearly 160,000 words but not even 6,000 annotated relation instances [13]. Other examples address relatively small corpora which contain only a few documents, e. g. twenty documents in [14].

In order to account ontological structures (vertical and horizontal taxonomies) in OBIE evaluations, common evaluation metrics such as *precision*, *recall*, and *f-measure* were extended to an *augmented precision and recall* [15].

For ontology-based annotation of text corpora, special tools are needed that respect both, the ontology's structure and instances, and the textual content. MAYNARD [6] gives benchmark criteria to assess such tools, namely interoperability, usability, accessibility, scalability, and reusability. With respect to these criteria, we used the Knowtator annotation tool [16].

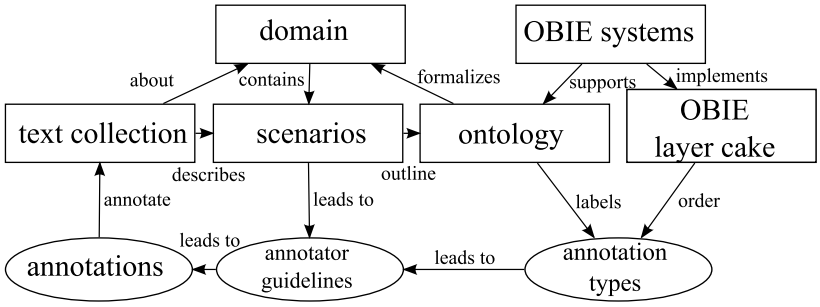

Fig. 1. OCAS: Scheme for creating an ontology-based, annotated corpus

In contrast to existing ontology-based gold corpora, our OCAS2008 gold standard is designed to be free to use for scientific purpose. Furthermore, it is completed, has a high density of annotations, and provides even implicit facts that are not explicitly contained in text. These facts can be inferred by humans given the underlying domain ontology at hand.

3 Ontology-Based Corpus and Annotation Scheme

With respect to corpus creation methodologies for traditional IE [10], the creation of an OBIE test corpus requires several steps.

First, design decisions about the domain and modularization have to be made. Second, an appropriate ontology for evaluation purpose and a text corpus have to be constructed and selected. This corpus is manually annotated in a third step. These annotations follow the modularization and refer to the ontology. In the last step the annotated text corpus has to be validated against predefined quality criteria.

We call this scheme the *Ontology-Based Corpus and Annotation Scheme (OCAS)*. Figure 1 summarizes intermediate steps inside the OCAS process. It uses an abstraction from concrete OBIE systems by applying the *OBIE Layer Cake* which is going to be discussed later. The following sections describe best practice approaches for those steps.

3.1 Design Decisions

We account an ontology-based annotated test corpus to be limited to a set of attributes namely *Closeness*, *Compactness*, and *Richness*. For our OCAS2008 test corpus we chose the Olympic Summer Games 2004 as domain and comment each attribute along with it.

Closeness An information domain is closed if it is limited to a few, but strictly defined topics. In terms of modeling a domain ontology, all instances of the domain can be defined.

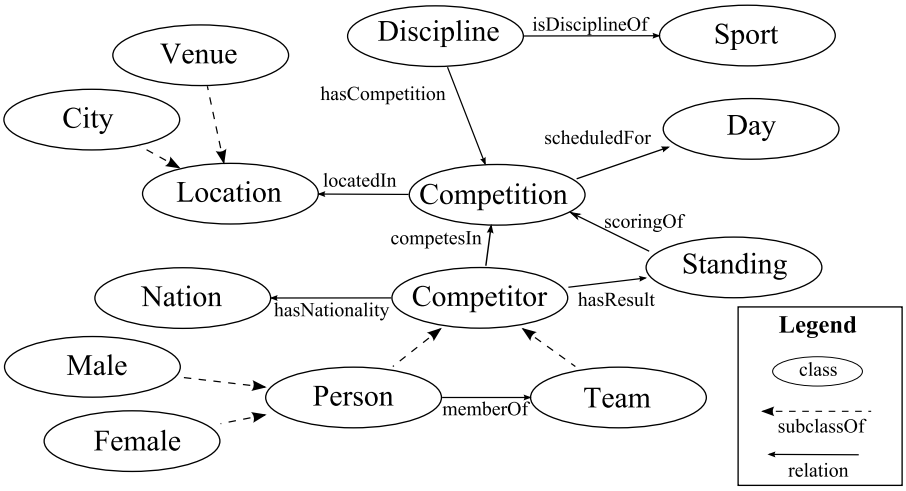

Fig. 2. The ontology for the OCAS2008 test corpus.

The domain of the Olympic Summer Games 2004 is closed with respect to this terminology: We know exactly which athletes, disciplines, events, etc. occurred during that Games and we can name them all. Note that all instances and facts could be gathered on Yahoo![5] and Wikipedia[6].

Compactness In a compact domain the key concepts modeling this domain are highly coherent. Thus, an ontology about this domain is also compact and contains only a few classes which share relations. In terms of a semantic graph, these relations should be linear independent in order to reduce redundancies. As a result just a few instances in a text passage share many relations and therefore describe many bits of knowledge. This ensures a high density of explicit and implicit facts (that are triples in the style of subject, predicate, object) in text sources.

The domain covering the Olympic Summer Games 2004 is compact, i. e. the created ontology consists of nine concrete classes with nine relations (see Fig. 2). The maximum distance between two ontology nodes is four. The domain contains more than 40,000 relevant instances. (Around 11,988 athletes participated.)

Richness A populated ontological domain which can be covered completely by a text corpus allows significant evaluations, i. e. it is said to be rich. Such a domain ontology contains a large amount of instances. Additionally this also allows to assess the scalability of an OBIE system.

The Olympic Summer Games 2004 are a rich domain in this sense. Plenty of news articles exist about each olympic summer games.

Many online news providers are available concerning the Olympic Summer Games 2004 which was crucial for our decision. This rich source allowed an easy retrieval of 121 news articles from ABC[7] and BBC[8], as described later.

[5] `http://sports.yahoo.com/olympics/athens2004`
[6] `http://en.wikipedia.org/w/index.php?title=2004_Summer_Olympics&oldid=221044690`
[7] `http://www.abc.net.au/olympics/2004/news_archive.htm`
[8] `http://news.bbc.co.uk/sport2/hi/olympics_2004/`

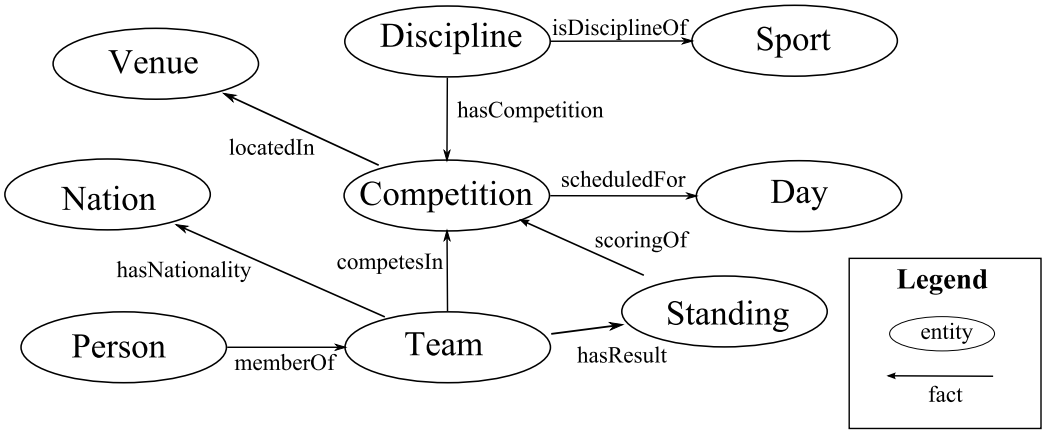

Fig. 3. The team-based scenario map for the OCAS2008 annotation process.

Real-life domains are likely to not respect these attributes. But for evaluation purposes – and the created gold standard is meant to allow such evaluation – this limitations seem appropriate.

3.2 Ontology and Text Corpus

Based on the selected domain, a sufficient amount of domain related text documents has to be retrieved. The individual documents should be representative for the domain. Moreover they should be selected in a transparent way without any influence from IE developers. Following this, we decided to choose OCAS2008 documents by using the information retrieval system DynaQ[9] [17]. The text corpus should cover at least three documents about each olympic day (from August, 11 to August, 29) and at least two documents about each of the 32 olympic disciplines. At first, we indexed 5000 news article from ABC and BBC with DynaQ. Then, we requested the best matching documents for each olympic discipline by querying for the discipline's name. For each query, we chose the two best fitting documents. DynaQ offers the feature to define a set of documents as context and search for similar matches. We defined the resulting number of 64 documents as context and grouped the resulting set by day. For each day from August, 11 to August, 29 we took the three best fitting results that were not already inside the context set. This method ensured a high variety and coverage of text.

Before designing a suitable ontology we created so called *scenario maps*. These maps define a scenario with the relevant types of entities and their relations that occur in one message type of text. In our domain we observed that news articles usually describe results of either single athletes or whole teams. Thus we defined two scenario maps. Figure 3 shows one scenario map defining team results. (e.g., several persons (athletes) are member of a team, that competes in a certain competition, that is scheduled for a day . . .)

[9] `http://dynaq.opendfki.de`

Scenario maps specify a desired annotation *goal* in a descriptive way. They act as an instrument of quality assurance in terms of annotation completeness and relevancy. Apart from entity annotations, scenario maps specify desired facts inside a scenario that are contained in every appropriate text, either explicitly or implicitly. For example, the information at which venue a specific competition took place is rarely stated in a news article. But an expert in our domain would know this fact implicitly. Therefore, the annotation quality can be assessed by checking whether an annotated document includes the required scenario annotations or not.

We modeled the domain ontology after constructing the scenario maps in a *scenario-based* way. That means the ontology in Fig. 2 evolves from the scenario map in Fig. 3 and others. The resulting domain ontology is an aggregate of all scenario maps that were defined about the domain. This approach ensures that the ontology is most suitable for the selected evaluation domain and the annotation process.

In order to keep evaluation simple, structures of evaluation ontologies should on the one hand be as small and lightweight as possible while on the other hand covering all necessary entities within the domain. In order to stay realistic, the hierarchy should contain some deep as well as some shallow parts. In a second step the ontology has to be populated. A population with a complete set of instances for the selected domain finally leads to an easier annotation process, as known instances can be annotated by a corresponding *Unique Resource Identifier (URI)*. In Semantic Web standards[10], such as OWL or RDF, it is common to identify instances by URI. Common OBIE or annotation systems use such Semantic Web standards to model their domain ontologies [5], [18], [4], [3]. Thus, we based our OCAS2008 process on these standards, e. g. we modeled instantiated facts as RDF triples in style of subject, predicate (relation), and object.

One important criterion is that all ontology classes are annotated in the corpus. A too big ontology can be reduced by the classes which are not annotated. To assure such completeness, the annotation scenarios are reused as annotators' checklists. Therefore, the available text must be classified according to the scenario type, e. g. we distinguished between news articles reporting results of whole teams or single athletes.

3.3 Annotation

Similar to traditional IE, OBIE may also be divided into subtasks [14]. Adopting HOBBS, we structured these OBIE tasks in different layers and called it *OBIE layer cake*. Each layer can be matched by at least one *annotation type*. Comparing evaluation results from different OBIE systems is now possible by mapping each system's layers to corresponding annotation types. This ensures a comparable benchmarking between different OBIE systems. Figure 4 gives an overview of the OBIE layer cake and the corresponding five annotation types. The two gray colored lower layers *Normalization* and *Segmentation* focus on correct text extraction from proprietary document formats and correct recognition of text segments such as tokens, sentences, or paragraphs. These rather syntactic tasks form the base for any text-based analysis. But as OBIE focuses on succeeding semantic analysis, we focus on the upper three annotation layers tagged

[10] http://www.w3.org/2001/sw/

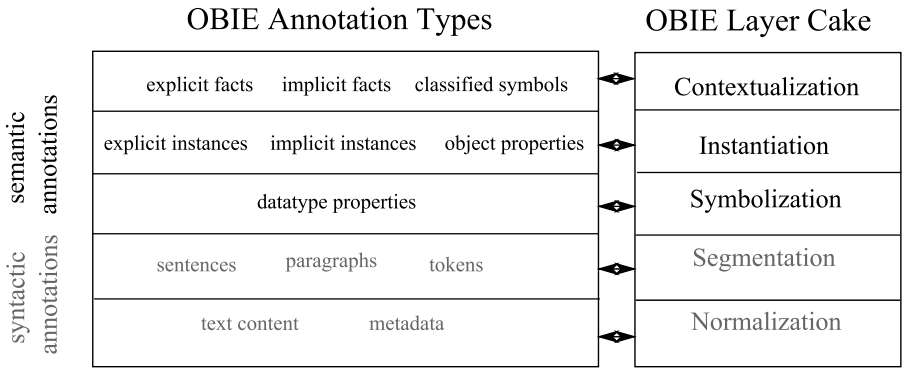

Fig. 4. OBIE Layer Cake: Layered annotation types in relation to abstract OBIE tasks.

as semantic annotations. They belong to the layers *Symbolization*, *Instantiation*, and *Contextualization*.

Symbolization contains annotations about tokens that match current datatype properties of the underlying ontology. In terms of OCAS2008 and given the datatype properties `firstName`, `lastName`, and `name`, the sentence *"Ibrahim Abdul Razak plays for Ghana"* may be annotated with `names` in terms of *"Ibrahim Abdul Razak"*, *"plays for"*, and *"Ghana"*. *Ibrahim* is annotated with `firstName` and *"Abdul Razak"* is annotated with `lastName`. *"Ghana"* and *"plays for"* are annotated as `name`. They are names of a country and a relation respectively.

Instantiation contains annotations about concrete instances and object relations that have been resolved from symbols. Given the sentence *"Ibrahim Abdul Razak plays for Ghana"* and the symbols mentioned above results in two instances (the person `urn:Ibrahim+Abdul+Razak` and the nation `urn:Ghana`) and one object property `urn:hasNationality`. These annotations are concerned to be explicit as they refer to symbols present in the text. An implicit instance in this context may be the instance `urn:Soccer`, i. e. the ontology knows that Ibrahim Abdul Razak plays soccer. Technically spoken, implicit instances are those annotated instances that are defined in a scenario but do not occur in a certain text.

Contextualization contains annotations about facts as well as not instantiated but classified symbols. Given the sentence *"Ibrahim Abdul Razak plays for Ghana"* and the above defined annotations results in the fact in style of a triple (`urn:Ibrahim+Abdul+Razak`, `hasNationality`, `urn:Ghana`).

An implicit fact may be (`urn:Ibrahim+Abdul+Razak`, `urn:memberOf`, `urn:Soccer+Team+Ghana`). Here, the instance `urn:Soccer+Team+Ghana` is not annotated in the text document, but is part of our domain ontology. Technically spoken implicit facts are those annotated facts that is based on at least one implicit instance in subject, predicate, or object.

Annotations of types between different layers are connected by resolution ordering called indication. This means that symbols are resolved as instances properties (relations or facets). Instances and properties build facts.

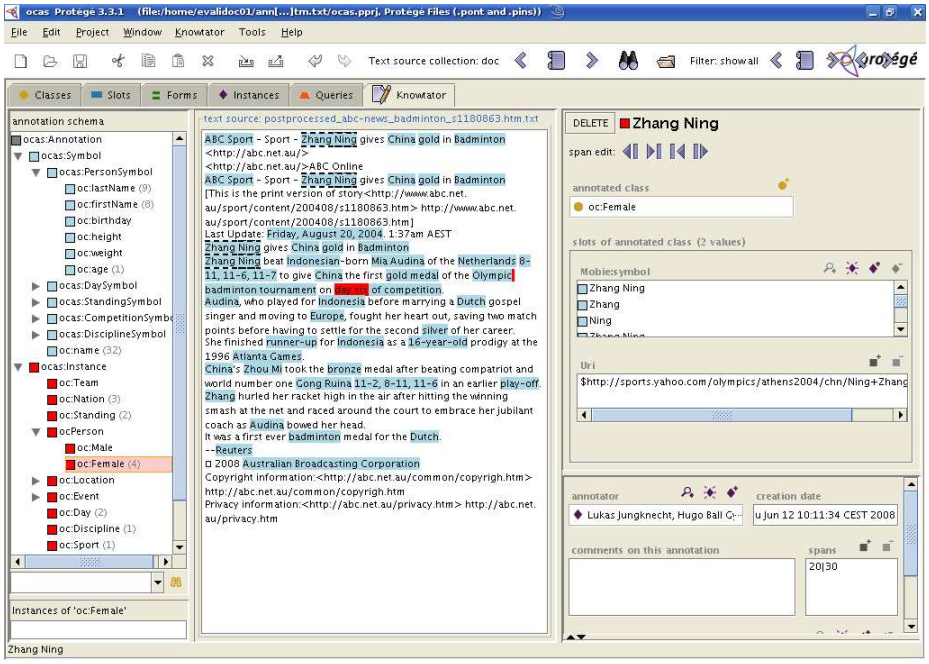

Fig. 5. Screenshot of the Knowtator plugin with our different annotation types labeled according to our domain ontology.

In our approach, we employ *Knowtator* [16] for manual annotation. It allows us to model concrete annotation types using the ontology editor *Protégé*[11]. Figure 5 shows a screenshot of the Protégé plugin Knowtator with the focus on an annotated text. The different annotation types are grouped on the left side. These types are marked with different colors, serving as a visual aid for annotators.

As outlined in Fig. 1 scenario maps and annotation types lead to so-called *annotator guidelines*. While scenario maps define a declarative annotation goal, annotation types and indications in between define annotation operations for reaching this goal. Annotator guidelines are compulsory guidelines for annotators defining the scenario to apply and the order in which a text has to be annotated with types of annotations for completing the scenario. We achieved two results by using these guidelines during our annotation process:

– We ensured that a considerable amount of implicit facts was annotated, by applying scenarios.

[11] `http://protege.stanford.edu`

– The given order in which the different annotation types where annotated aided annotators, i. e. as symbols where annotated first, the annotation of instances in the second step became easier: Every possible instance was annotated as symbol before.

High quality annotator guidelines minimize annotator's callbacks during the annotation process and reduce ambiguities in annotating text. In addition, annotators that follow these guidelines ensure consistent resolution traces between layered annotations.

3.4 Corpus Validation

We used the previously presented methods for annotating 121 selected documents. They contained a total of 31,102 words and were annotated by six people during eight days. The annotators were high school students and had no previous knowledge in the field of IE. The annotation process took a total of 176 person hours. This first analysis results in a cost estimation of 5.66 hours per annotation of 1,000 words. This does not include the necessary preparation and post processing. Further analysis must be conducted as other measures would also be interesting: The average distribution of different annotation types or concepts in the annotated documents is just one example. Prior to conducting this analysis we plan three validation steps according to the following quality criteria:

Completeness Checking whether the documents were annotated *completely* with respect to our scenario maps.

Consistency Test the *consistency* of the annotations regarding our annotation types, e. g. each annotated instance must also be marked as symbol.

Correctness Verify the *correctness* of annotations, i. e. if every annotated instance in the corpus respects our domain ontology's instance set and uses the same URIs.

These quality criteria were designed to limit the evaluators' actions for modifying the objectivity of a corpus. Other activities might lead to a biased and subjective change in the corpus and finally affect evaluation results.

4 Evaluation

In the near future the gold standard OCAS2008 is going to be used for evaluating and comparing the OBIE systems GATE and iDocument[12] along the OBIE tasks *Symbolization*, *Instantiation*, and *Contextualization*. The results will be free and presented online at `http://idocument.opendfki.de`.

5 Summary and Outlook

In this work we described an approach for creating a semantically annotated corpus in order to evaluate OBIE systems. We commented the creation process with best practices according to state-of-the-art corpus creation methodologies and finally our own experiences.

[12] `http://idocument.opendfki.de`

We considered three corpus requirements namely *closeness*, *compactness*, and *richness*. We account to validate a corpus' annotations by considering three quality criteria, namely *completeness*, *consistency*, and *correctness*. In order to compare multiple OBIE systems based on OCAS2008, we provide a generic view on OBIE systems called *OBIE layer cake*. In addition the OCAS2008 gold standard provides:

- An OBIE layer cake for comparing OBIE systems by subtasks,
- a document corpus of 121 documents with 31,000 words about a completed domain,
- a complete domain ontology including more than 40,000 instances,
- two annotation scenarios that extend traditional template-based evaluations,
- an annotation set that contains typed annotations according to the ontology and the OBIE layer cake,
- annotations that concern text segments, symbols, instances, explicitly written facts, implicit facts, and
- finally, human created annotations according to predefined specifications.

Further future activities comprise a detailed validation of the OCAS2008 corpus along closeness, consistency, and correctness. This validation step is of crucial importance for the assessment of a corpus' value and the significance of later evaluation results. Only with a corpus exceeding specific qualitative standards meaningful results can be get.

After the validation process and conducting several statistic analysis considering annotation distributions, OCAS2008 is planned to be free to use for scientific purposes. Amongst others our ontology, a description of our Information Retrieval system for selecting the 121 text documents, and the documents with annotations will be available at OpenDFKI[13]. Finally, an evaluation of the OBIE system iDocument and GATE is planned.

The creation of OCAS2008 was expensive. Only the annotation process itself took about 176 person hours. As an example let us assume that student annotators can be recruited for the annotation process. With an hourly rate of € 11 this leads to costs of € 62 per 1,000 words of semantically annotated text. Additional work has to be done also, which increases costs further. Another problem occurs for more complex domains which need experts, who are more expensive than students, to annotate the corpus. It is still an open question how to decrease costs for corpus creation.

6 Acknowledgement

Thanks to our tough annotators Jan Adamczyk, Patrick Hütchen, Lukas Jungknecht, Michel-Hardy Kling, Ewald Leibham, and Franz Reck. This work was supported by "Stiftung Rheinland-Pfalz für Innovation".

[13] `https://idocument.opendfki.de`. Free registration may be required.

References

1. Hobbs, J.R.: The generic information extraction system. In: MUC5 '93: Proceedings of the 5th conference on Message understanding, Morristown, NJ, USA, ACL (1993) 87–91
2. Sintek, M., Junker, M., van Elst, L., Abecker, A.: Using Information Extraction Rules for Extending Domain Ontologies. In: Workshop on Ontology Learning. CEUR-WS.org (2001)
3. Bontcheva, K., Tablan, V., Maynard, D., Cunningham, H.: Evolving GATE to Meet New Challenges in Language Engineering. Natural Language Engineering **10**(3/4) (2004) 349—373
4. Maedche, A., Neumann, G., Staab, S.: Bootstrapping an Ontology-based Information Extraction System. Studies in Fuzziness and Soft Computing. In Szczepaniak, P., Segovia, J., Kacprzyk, J., Zadeh, L.A., eds.: Intelligent Exploration of the Web. Springer, Berlin (2002)
5. Buitelaar, P., Cimiano, P., Racioppa, S., Siegel, M.: Ontology-based Information Extraction with SOBA. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC), ELRA (MAY 2006) 2321–2324
6. Maynard, D.: Benchmarking ontology-based annotation tools for the Semantic Web. In: In UK e-Science Programme All Hands Meeting (AHM2005) Workshop Text Mining, e-Research and Grid-enabled Language Technology. (2005)
7. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A framework and graphical development environment for robust NLP tools and applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics. (2002)
8. Chinchor, N.: Overview of MUC-7/MET-2. In: Message Understanding Conference Proceedings: MUC-7. (1998)
9. NIST: ACE08 Evaluation Plan. `http://www.nist.gov/speech/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf` (2008)
10. Linguistic Data Consortium, University of Pennsylvania: Creating Data Resources. http://www.ldc.upenn.edu/Creating (2007)
11. Makhoul, J., Kubala, F., Schwartz, R., Weischedel, R.: Performance measures for information extraction. In: Proceedings of DARPA Broadcast News Workshop. (1999) 249–252
12. Peters, W., Aswani, N., Bontcheva, K., Cunningham, H.: Quantitativa Evaluation Tools and Corpora V1. Technical report, SEKT project deliverable D2.5.1 (2005)
13. Wang, T., Li, Y., Bontcheva, K., Cunningham, H., Wang, J.: Automatic Extraction of Hierarchical Relations from Text. In: ESWC. (2006) 215–229
14. Adrian, B., Dengel, A.: Believing Finite-state cascades in Knowledge-based Information Extraction. In: KI 2008: Advances in Artificial Intelligence. (2008, to appear)
15. Maynard, D., Peters, W., Li, Y.: Metrics for Evaluation of Ontology-based Information Extraction. In: Proceedings of WWW 2006 Workshop on Evaluation of Ontologies for the Web (EON 2006). (2006)
16. Ogren, P.V.: Knowtator: A Protege plug-in for annotated corpus construction. In: Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume: demonstrations. (2006) 273–275
17. Agne, S., Reuschling, C., Dengel, A.: Dynaq - dynamic queries for electronic document management. In: EDOCW '06: Proceedings of the 10th IEEE on International Enterprise Distributed Object Computing Conference Workshops, Washington, DC, USA, IEEE Computer Society (2006) 61
18. Cimiano, P., Handschuh, S., Staab, S.: Towards the self-annotating web. In: WWW '04: Proceedings of the 13th international conference on World Wide Web, New York, NY, USA, ACM (2004) 462–471