Text Mining through Semi Automatic Semantic Annotation

Nadzeya Kiyavitskaya[1], Nicola Zeni[1], Luisa Mich[2],
James R. Cordy[3], and John Mylopoulos[4]

[1] Dept. of Information and Communication Technology, University of Trento, Italy
{nadzeya, nzeni}@dit.unitn.it
[2] Dept. of Computer and Management Sciences, University of Trento, Italy
luisa.mich@unitn.it
[3] School of Computing, Queens University, Kingston, Canada
cordy@cs.queensu.ca
[4] Dept. of Computer Science, University of Toronto, Ontario, Canada
jm@cs.toronto.edu

Abstract. The Web is the greatest information source in human history. Unfortunately, mining knowledge out of this source is a laborious and error-prone task. Many researchers believe that a solution to the problem can be founded on semantic annotations that need to be inserted in web-based documents and guide information extraction and knowledge mining. In this paper, we further elaborate a tool-supported process for semantic annotation of documents based on techniques and technologies traditionally used in software analysis and reverse engineering for large-scale legacy code bases. The outcomes of the paper include an experimental evaluation framework and empirical results based on two case studies adopted from the Tourism sector. The conclusions suggest that our approach can facilitate the semi-automatic annotation of large document bases.

Keywords: semantic annotation, large-scale document analysis, conceptual schemas, software analysis.

1 Introduction

The Web is the greatest information source in human history. Unfortunately, mining knowledge out of this source is a laborious and error-prone task, much like looking for the proverbial needle in a haystack. Many researchers believe that a solution to the problem can be founded on semantic annotations that need to be inserted in web-based documents and guide information extraction and knowledge mining. Such annotations use terms defined in an ontology. We are interested in knowledge mining the Web, and use semantic annotations as the key idea in terms of which the mining is to be done.

However, adding semantic annotations to documents is also a laborious and error-prone task. To help the annotator, we are developing tools that facilitate the annotation process by making a first pass at the documents, inserting annotations on the basis of textual patterns. The annotator can then make a second pass improving manually the annotations. The main objective of this paper is to present a tool-supported methodology that semi-automates the semantic annotation process for a set of documents with respect to a semantic model (ontology or conceptual schema). In this work we propose to approach the problem using highly efficient methods and tools proven effective in the software analysis domain for processing billions of lines of legacy software source code [2]. In fact, document analysis for the Semantic Web and software code analysis have striking similarities in their needs:

- robust parsing techniques, given that real documents rarely match given grammars;
- a semantic understanding of source text, on the basis of a semantic model;
- semantic clues drawn from a vocabulary associated with the semantic model;
- contextual clues drawn from the syntactic structure of the source text;
- inferred semantics from exploring relationships between identified semantic entities and their properties, contexts and related other entities.

On the basis of these considerations, we have adapted software analysis techniques to the more general problem of semantic annotation of text documents. Our initial hypothesis is that these methods can attain the same scalability for analysis of textual documents as for software code analysis. In this work we extend and generalize the process and architecture of the prototype semantic annotation tool presented earlier in [3]. The contribution of this work includes also an evaluation framework for semantic annotation tools, as well as two real-world case studies: accommodation advertisements and Tourist Board web sites. For the first experiment, we use a small conceptual schema derived from a set of user queries. For the second experiment, we adopt more elaborated conceptual schemas reflecting a richer semantic domain.

Our evaluation of both applications uses a three-stage evaluation framework which takes into account:

- standard accuracy measures, such as Recall, Precision, and F-measure;
- productivity, i.e. the fraction of time spent for annotation when the human is assisted by our tool vs. time spent for manual annotation “from scratch”; and
- a calibration technique which recognizes that there is no such thing as “correct” and “wrong” annotations, as human annotators also differ among themselves on how to annotate a given document.

The rest of the paper is organized as follows. Our proposed annotation process and the architecture of our semantic annotation system are introduced in section 2. The two case studies are presented in section 3, and section 4 describes the evaluation setup and experimental results. Section 5 provides a short comparative overview of semantic annotation tools and conclusions are drawn in section 6.

2 Methodology

Our method for semantic annotation of documents uses the generalized parsing and structural transformation system TXL [4], the basis of the automated Year 2000 system LS/2000 [5]. TXL is a programming language specially designed to allow by-example rapid prototyping of language descriptions, tools and applications. The system accepts as input a grammar and a document, generates a parse tree for the input document, and applies transformation rules to generate output in a target format. The architecture of our solution (Fig. 1) is based on the LS/2000 software analysis architecture, generalized to allow for easy parameterization by a range of semantic domains.

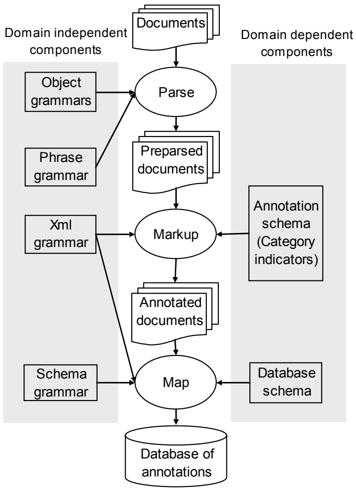

Fig. 1. Architecture of our semantic annotation process.

The architecture explicitly factors out reusable domain independent knowledge such as the structure of basic entities (email and web addresses, dates, and other word-equivalent objects) and language structures (document, paragraph, sentence and phrase structure), shown on the left hand side, while allowing for easy change of semantic domain, characterized by vocabulary (category word and phrase lists) and semantic model (entity-relationship schema and interpretation), shown on the right.

The process consists of three phases. In the first stage, an approximate ambiguous context-free grammar is used to efficiently obtain an approximate phrase structure parse of the source text using the TXL parsing engine. Using robust parsing techniques borrowed from compiler technology [6] this stage results in a deterministic maximal parse. As part of this first stage, basic entities are recognized. The parse is linear in the length of the input and runs at compiler speeds.

In the second stage, initial semantic annotation of the document is derived using a wordlist file specifying both positive and negative indicators for semantic categories. Indicators can be both literal words and phrases and names of parsed entities.

Phrases are marked up once for each category they match – thus at this stage a sentence or phrase may end up with many different semantic markups. Vocabulary lists are derived from the semantic model for the target domain. This stage uses the structural pattern matching and source transformation capabilities of the TXL engine similarly as for software markup to yield a preliminary marked-up text in XML form.

The third stage uses the XML marked-up text to populate an XML database schema, derived from the semantic model for the target domain. Sentences and phrases with multiple markups are "cloned" using TXL source transformation to appear as multiple copies, one for each different markup, before populating the database. In this way we do not prejudice one interpretation as being preferred.

The outputs of our process are both the XML marked-up text and the populated database. The populated database can be queried by a standard SQL database engine.

3 Experimental Case Studies

Our case studies involve two applications in the Tourism area. Tourism is a very broad sector of economy which comprises many heterogeneous domains: accommodation and eating structures, sports, means of transport, historical sites, tourist attractions, medical services and other areas of human activity. Information available from heterogeneous data sources must be integrated in order to allow effective interoperability of tourism information systems and to enable knowledge mining for the variety of roles and services that characterize such a compound sector (e.g. composition of services for tourist packages). This is where semantic annotations come in handy.

3.1 Accommodation Ads

As a first full experiment in the application of our new method, we have been working in the domain of travel documents, and in particular with published advertisements for accommodation. This domain is typical of the travel domain in general and poses many problems commonly found in other text markup problems, such as: partial and malformed sentences; abbreviations and short-forms; location-dependent vocabulary; monetary units; date and time conventions, and so on.

In the first case study we used a set of several hundred advertisements for accommodation in Rome drawn from an online newspaper. The task was to identify and mark up the categories of semantic information in the advertisements according to a given accommodation conceptual schema (Fig. 2), which was reduced by hand to an XML schema for input to our system. The desired result was a database with one instance of the schema for each advertisement in the input, and the marked-up original advertisements. To adapt our semantic annotation methodology to this experiment the domain-related wordlists were constructed by hand from a set of examples.

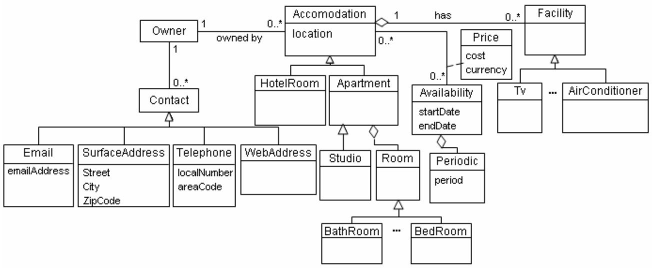

Fig. 2. Conceptual schema for accommodation ads.

3.2 Tourist Board Web Pages

In the second case study we pursued two main goals: to demonstrate the generality of our method over different domains, and to verify the scalability of our approach on a richer semantic model and larger natural language documents. For this purpose, we considered the web sites of Tourist Boards in the province of Trentino (Italy)[1] as input documents. In contrast to the classified ads, this domain presents a number of specific problematic issues: free unrestricted vocabulary; differently structured text; a rich semantic model covering the content of web sites.

This experiment was run in the collaboration with the marketing experts of the eTourism[2] group of University of Trento. From the point of view of tourist marketing experts in tourism, the high-level business goal of this case study was to assess the communicative efficacy of the web sites based on content quality or *informativity*, that is, how comprehensively the web site covers relevant topics according to the strategic goals of the Tourist Board.

In order to assess the communicative efficacy we performed semantic annotation of the web pages revealing the presence of information important for a Tourist Board web site to be effective. The list of semantic categories and their descriptions was provided by the tourism experts (Fig. 3).

- **Geography**
  - Climate
  - Weather predictions
  - Land Formation
  - Lakes and Rivers
  - Landscape
- **Local products**
  - Local handcrafting
  - Agricultural products
  - Gastronomy
- **Culture**
  - Traditions and customs
  - Local history
  - Festivals
  - Population
  - Cultural institutions and associations
  - Libraries
  - Cinemas
  - Local literature
  - Local prominent people
- **Artistic Heritage**
  - Places to visit: museums, castles
  - Tickets, entrance fees, guides
- **Sport**
  - Sporting events
  - Sport infrastructure
  - Sport disciplines
- **Accommodation**
  - Places to stay
  - How to book
  - How to arrive
  - Prices
  - Availability
- **Food and refreshment**
  - Places to eat
  - Dishes
  - Degustation
  - Time tables
  - How to book
- **Wellness**
  - Wellness centers
  - Wellness services
- **Services**
  - Transport, schedules
  - Information offices
  - Terminals, stations, airports
  - Travel agencies

Fig. 3. Relevant categories for communicative efficacy of a Tourist Board web site.

[1] http://www.trentino.to/home/about.html?_area=about&_lang=it&_m=apt

[2] http://www.economia.unitn.it/etourism

In this second experiment, we adapted our annotation framework to the new domain by replacing the domain-dependent components with respect to this specific task. For this purpose, the initial rough schema provided by the domain experts was transformed into a richer conceptual schema consisting of about 130 concepts systematized into a hierarchy and connected by semantic relations (see the partial view in Fig. 4[3]).

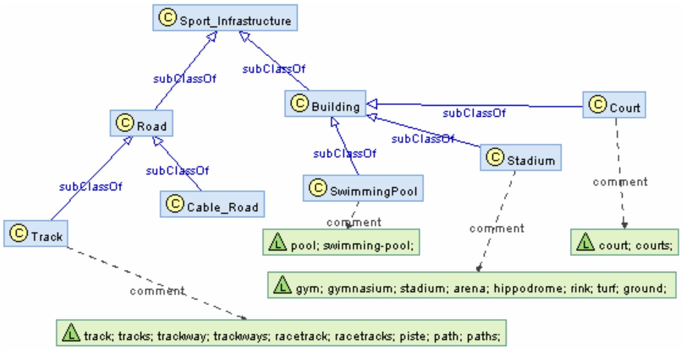

Fig. 4. A slice of the conceptual schema showing semantic (placement in the hierarchy, relationships, attributes) and syntactic (keywords or patterns) information associated with concepts. This view shows only *is-a* relations, because this type of relation is essential in guiding the annotation process. The complete model includes many more relations apart from taxonomical ones.

Domain dependent vocabulary was derived semi-automatically, expanding concept definitions with the synonyms provided by the *WordNet*[4] database and on-line Thesaurus[5] and mined from a set of sample documents. The total number of keywords collected was 507 and an additional four object patterns were re-used from previous application to detect such entities as monetary amounts, e-mails, web addresses and phone numbers.

To begin this experiment we downloaded the English version of 13 Tourist Board web sites using an offline browser software[6]. For some of them (which are generated dynamically) we had to apply a manual screen-scraping technique. Then two human annotators and the tool were given 11742 text fragments for annotation. The required result was a database with one instance of the schema for each Tourist Board web site, and the marked-up original text (Fig. 5).

```
   <FoodAndRefreshment>Bread and wine snack in the shade of an elegant
park.</FoodAndRefreshment>
   <FoodAndRefreshment>Dinner at the "La Luna Piena" restaurant,
consisting of the "Il Piatto del Vellutaio"</FoodAndRefreshment>
   <ArtisticHeritage>Museo del Pianoforte Antico: guided visit and
concert proposed within the "Museum Nights" programme on the 3, 10, 17
and 24 of August.</ArtisticHeritage>
```

Fig. 5. Example of XML-marked up content of a tourism web site.

[3] The visualization tool RDFGravity: http://semweb.salzburgresearch.at/apps/rdf-gravity/

[4] http://wordnet.princeton.edu

[5] http://thesaurus.reference.com

[6] SurfOffline 1.4: http://www.bimesoft.com

4 Experimental Evaluation

4.1 Evaluation Framework

The performance of semantic annotation tools is usually evaluated similarly to information extraction systems, i.e. by comparing with a reference correct markup and calculating recall and precision metrics.

In order to evaluate our initial experimental results, we designed a three stage validation process. At each stage, we calculated a number of metrics [7] for the tool's automated markup compared to manually-generated annotations: *Recall* evaluates how well the tool performs in finding relevant items; *Precision* shows how well the tool performs in not returning irrelevant items; *Fallout* measures how quickly precision drops as recall is increased; *Accuracy* measures how well the tool identifies relevant items and rejects irrelevant ones; *Error rate* demonstrates how much the tool is prone to accept irrelevant items and reject relevant ones; *F-measure* is an harmonic mean of recall and precision.

In the first step of our evaluation framework, we compare the system output directly with manual annotations. We expect that quality of manual annotations constitutes an upper bound for automatic document analysis. Of course, this type of evaluation can't be applied on a large scale for cost reasons.

In the second step, we check if the use of automatic tool increases the productivity of human annotators. We note the time used for manual annotation of the original textual documents and compared it to the time used for manual correction of the automatically annotated documents. The percentage difference of these two measures shows how much time can be saved when the tool assists the human annotator.

Finally, in our third step we take into account disagreement between annotators to interpreted the automatically obtained annotation. Then, we compare system results against the final human markup made by correcting the automatically generated markup.

4.2 Experimental Results

Experiment 1: Accommodation Ads. The details of our evaluation for the accommodation ads application can be found in [2]. We only say that as a result of this first experiment, even without local knowledge and using a very small vocabulary and only few TXL rules, we obtained results comparable to some of the best heavyweight methods, albeit on a very limited domain. Performance of our untuned experimental tool was also already very fast, handling for example 100 advertisements in about 1 second on a 1 GHz PC.

Experiment 2: Tourist Board Web Pages. As the semantic model in this experiment was fairly extensive, we could not afford humans to handle properly all of the entities of the rich domain schema. Accordingly, in our evaluation we considered only general categories in the annotation schema (*Geography, Sport, Culture, Artistic Heritage, Local Products, Wellness, Accommodation, Food and Refreshment,* *Services*). For these we performed simple metrics-based validation (Tables 1a, b, c) and calibration of the results taking into account inter-annotator disagreement (Table 2) for the entire set of 11742 paragraphs.

Table 1a. Evaluating system annotation vs. human Annotator 1.

Topic Measure	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	68.23	68.18	72.49	82.28	82.57	83.19	68.29	16.67	76.42
Precision	85.62	82.19	93.16	97.38	78.35	96.12	94.92	50.00	91.01
Fallout	0.59	0.34	0.50	0.19	1.50	0.11	0.08	0.03	0.43
Accuracy	97.88	98.95	97.16	98.39	97.52	99.39	99.26	99.85	98.31
Error	2.12	1.05	2.84	1.61	2.48	0.61	0.74	0.15	1.69
F-measure	75.94	74.53	81.53	89.19	80.40	89.19	79.43	25.00	83.08

Table 1b. Evaluating system annotation vs. human Annotator 2.

Topic Measure	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	42.19	59.09	74.85	70.57	73.86	68.91	40.24	16.67	59.43
Precision	69.83	82.54	59.81	59.31	62.24	50.62	55.93	33.33	33.96
Fallout	0.94	0.29	4.75	4.25	2.94	2.11	0.68	0.05	6.62
Accuracy	96.27	98.80	93.48	93.71	95.63	97.01	98.08	99.82	91.54
Error	3.73	1.20	6.52	6.29	4.37	2.99	1.92	0.18	8.46
F-measure	52.60	68.87	66.49	64.45	67.55	58.36	46.81	22.22	43.22

Table 1c. Evaluating system annotation vs. humans – average category scores.

Measure	Tool vs. A1	Tool vs. A2
Recall	68.70	56.20
Precision	85.42	56.40
Fallout	0.42	2.51
Accuracy	98.52	96.04
Error	1.48	3.96
F-measure	75.37	54.51

Table 2. Comparing system results vs. human annotators.

Measure	A2 vs. A1	Tool vs. A1	A1 vs. A2	Tool vs. A2
Recall	61.75	68.70	76.47	56.20
Precision	76.47	85.42	61.75	56.40
Fallout	1.00	0.42	2.50	2.51
Accuracy	96.70	98.52	96.70	96.04
Error	3.30	1.48	3.30	3.96
F-measure	66.79	75.37	66.79	54.51

As shown in Table 2, for the given annotation schema the task turned out to be difficult both for the system and for the humans due to the vague definitions of the semantic categories. For example, text about local food may be associated with either or both of the *Local Products* category and the *Food and Refreshment* category, depending on the context. Explicit resolution of such ambiguities in the expert definition would improve the results. Interpreting the results of this case study, we must take into account also that the diversity in accuracy metrics is partially caused by the different experience of the annotators in the tourism area. If we compare the difference in scores of F-measure, as the most aggregate characteristic, the overall difference in performances of the system and the humans is approximately 10%.

In the second stage of evaluation, the human annotators were observed to use 72% less time to correct automatically annotated text than they spent on their original unassisted annotations.

In the third stage, when the human annotators corrected automatically marked up documents, the results of comparison to the final human markup are given in Tables 3a, b, c and calibration to human performance in Table 4.

Table 3a. Evaluating system annotation vs. human Annotator 1 as assisted by the tool.

Topic **Measure**	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	96.88	94.32	97.34	96.91	96.68	94.96	90.24	83.33	93.36
Precision	100.00	93.26	98.50	100.00	83.21	99.12	100.00	100.00	96.10
Fallout	0.00	0.16	0.14	0.00	1.28	0.03	0.00	0.00	0.22
Accuracy	99.85	99.72	99.64	99.74	98.59	99.82	99.80	99.97	99.44
Error	0.15	0.28	0.36	0.26	1.41	0.18	0.20	0.03	0.56
F-measure	98.41	93.79	97.92	98.43	89.44	97.00	94.87	90.91	94.71

Table 3b. Evaluating system annotation vs. human Annotators2 as assisted by the tool.

Topic **Measure**	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	100.00	97.73	99.11	100.00	99.17	99.16	100.00	66.67	98.10
Precision	94.58	97.73	90.79	73.14	84.45	72.39	89.13	80.00	92.41
Fallout	0.30	0.05	0.95	3.31	1.20	1.19	0.26	0.03	0.46
Accuracy	99.72	99.90	99.05	96.96	98.82	98.82	99.74	99.92	99.46
Error	0.28	0.10	0.95	3.04	1.18	1.18	0.26	0.08	0.54
F-measure	97.22	97.73	94.77	84.49	91.22	83.69	94.25	72.73	95.17

Table 3c. Evaluating system annotation vs. humans – average scores.

Measure	**Tool vs. A1**	**Tool vs. A2**
Recall	93.78	95.55
Precision	96.69	86.07
Fallout	0.20	0.86
Accuracy	99.62	99.16
Error	0.38	0.84
F-measure	95.05	90.14

Table 4. Comparing system results vs. humans assisted by the tool.

Measure	A2 vs. A1	Tool vs. A1	A1 vs. A2	Tool vs. A2
Recall	80.99	93.78	92.54	95.55
Precision	92.54	96.69	80.99	86.07
Fallout	0.19	0.20	1.00	0.86
Accuracy	98.88	99.62	98.88	99.16
Error	1.12	0.38	1.12	0.84
F-measure	86.02	95.05	86.02	90.14

In contrast to the first experiment, this second case study was much more difficult to set up and evaluate than the first for the following reasons:

- Ambiguity in annotations: the large conceptual model of the domain is more difficult for usage as it allows ambiguities in interpretation.
- Difficulty in identifying fragments to be annotated: web documents contain various text structures such as tables, menu labels, free text and others.
- Size of the documents: in contrast to ads, which contained only a few sentences, the Web sites were of about 300 kbyte of text in HTML markup for each site.

However, in conclusion of this experiment we can say that our semantic annotation framework was able to demonstrate reasonable quality of results on the more general documents and the richer domain while maintaining fast performance.

5 Related Work

A number of tools have been shown to do well for various kinds of assisted or semi-automated semantic annotation of web content.

Text mining approaches usually use text itself as the basis for an analysis. For example, in [8] linguistic patterns and statistical methods are applied to discover a set of relevant terms for a document. Some tools combine data mining techniques with information extraction techniques and wrappers, as DiscoTEX [9].

SemTag [10] is an application that performs automated semantic tagging of large corpora. It tags large numbers of pages with terms from an ontology, using corpus statistics to improve the quality of tags. SemTag detects the occurrence of the entities in web pages and disambiguates them.

The KIM platform [11] is an application for automatic ontology-based named entities annotation, indexing and retrieval. In KIM, as well as in SemTag, semantic annotation is considered as the process of assigning to the entities in the test links to their semantic descriptions, provided by ontology. KIM performs recognition of named entities with respect to the ontology and is based on GATE[7].

Another tool that has been used on a large-scale is SCORE [12], which integrates several information extraction methods, including probabilistic, learning, and knowledge-based techniques, then combines the results from the different classifiers.

Our approach fundamentally differs from all these tools: it uses a lightweight robust context-free parse in place of linguistic analysis; our method does not have the

[7] General Architecture for Text Engineering: http://gate.ac.uk/

learning phase, instead it has to be tuned manually when being ported to a particular application, substituting or extending domain dependent components; and it does not necessarily require a knowledge base of known proper entities, rather it infers their existence from their structural and vocabulary context in the style of software analyzers. This advantage helps make our tool faster and less dependent on the additional knowledge sources.

Much of the work in the information extraction community is aimed at "rule learning", automating the creation of extraction patterns from previously tagged or semi-structured documents [13] and unsupervised extraction [14]. While learning issues are not addressed by our work, the application of patterns to documents is in many ways similar to our method, in particular the ontology-based method of Embley et al. [15]. The major differences lie in the implementation – whereas Embley's method relies primarily on regular expressions, our approach combines high-speed context-free robust parsing with simple word search.

Wrapper induction methods such as Stalker [16] and BWI [17] which try to infer patterns for marking the start and end points of fields to extract, also relate well to our work. When the learning stage is over and these methods are applied, their effect is quite similar to our results, identifying complete phrases related to the target concepts. However, our results are achieved in a fundamentally different way – by predicting start and end points using phrase parsing in advance rather than phrase induction afterwards. The biggest advantage of wrappers is that they need small amount of training data, but on the other hand they strongly rely on contextual clues and document structure. In contrast, our method uses context-independent parsing and does not require any strict input format.

6 Conclusions and Future Work

We have presented and evaluated a tool-supported process for the semantic annotation of web documents. The evaluation of our proposal included two case studies and the experimental results suggest good performance on the part of the semantic annotation tool. More importantly perhaps, the results suggest that productivity of a human annotator can increase substantially if the annotator works with the output of our tool, rather than conduct the annotation task manually. Our experiments also suggest that the tool is scalable when used with larger document sets. Apart from the experimental evaluation, we also consider the evaluation scheme itself as a novel contribution in that it measures not only the quality of the annotation, but also productivity improvements for human annotators. Our evaluation framework also takes into account inter-annotator disagreements to appropriately interpret the scores of the tool (since the human's performance is the upper bound for the automatic tool).

Our future research plans include tackling the problem of automatically generating inputs to the annotation process, such as object grammars and category keywords. We also propose to conduct experiments adapting other techniques used in software analysis to improve the quality of annotations and to accommodate different annotation granularities.

7 References

1. Isakowitz, T., Bieber, M., Vitali, F.: Web Information Systems. *Communications of the ACM*, Vol. 41(1) 78–80, 1998
2. Cordy, J., Dean, T., Malton, A., Schneider, K.: Source transformation in software engineering using the TXL transformation system. *Information and Software Technology Journal*, Vol. 44 (2002) 827–837
3. Kiyavitskaya, N., Zeni, N., Cordy, J.R., Mich, L., Mylopoulos, J.: Applying Software Analysis Technology to Lightweight Semantic Markup of Document Text. In *Proc. of Int. Conf. on Advances in Pattern Recognition (ICAPR 2005)*, Bath, UK, 2005, 590–600
4. Cordy, J.: TXL – a language for programming language tools and applications. In *Proc. of the 4th Int. Workshop on Language Descriptions, Tools and Applications*, Electronic Notes in Theoretical Computer Science, Vol. 110 (2004) 3–31
5. Dean, T., Cordy, J., Schneider, K., Malton, A.: Experience using design recovery techniques to transform legacy systems. In *Proc. 17 Int. Conf. on Software Maintenance,*2001, 622–631
6. Cordy, J., Schneider, K., Dean, T., Malton, A.: HSML: Design-directed source code hotspots. In *Proc. of the 9th Int. Workshop on Program Comprehension,* 2001, 145–154
7. Yang, Y.: An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval,* 1999, Vol. 1 (1/2) 67–88
8. R. Feldman, M. Fresko, H. Hirsh, Y. Aumann, O. Liphstat, Y. Schler, M. Rajman. Knowledge Management: A Text Mining Approach. In *Proc. of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM98)*, 29–30, 1998.
9. Nahm, U. Y.; Mooney, R. J.: Text Mining with Information Extraction. In *Proc. of the Spring Symposium on Mining Answers from Texts and Knowledge Bases*, Stanford/CA, 2002, 60–67.
10. Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., Jhingran, A., Kanungo, T., McCurley, K.S., Rajagopalan, S., Tomkins, A., Tomlin, J.A., Zien, J.Y.: A Case for Automated Large-Scale Semantic Annotation. *Journal of Web Semantics*, 2003, Vol. 1(1) 115–132
11. Kiryakov, A., Popov, B., Terziev, I., Manov, D., Ognyanoff, D.: Semantic Annotation, Indexing, and Retrieval. *Journal of Web Semantics*, 2005, Vol. 2(1), 49–79
12. Sheth, A., Bertram, C., Avant, D., Hammond, B., Kochut, K., Warke, Y.: Managing Semantic Content for the Web. *IEEE Internet Computing*, 2002, Vol. 6(4) 80–87
13. Nobata, C., Sekine, S.: Towards automatic acquisition of patterns for information extraction. In *Proc. of Int. Conf. on Computer Processing of Oriental Languages*, 1999
14. Etzioni, O., Cafarella, M.J., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence 165* (2005) 91–134
15. Wessman, A., Liddle, S.W., Embley, D.W.: A generalized framework for an ontology-based data-extraction system. In *Proc. of the 4th Int. Conf. on Information Systems Technology and its Applications* (2005) 239–253
16. Muslea, I., Minton, S., Knoblock, C.A.: Active learning with strong and weak views: A case study on wrapper induction. In *Proc. of the 18th Int. Joint Conf. on Artificial Intelligence (*2003) 415–420
17. Freitag, D., Kushmerick, N.: Boosted wrapper induction. In *Proc. of the 17th National Conf. on Artificial Intelligence* (2000) 577–583