An Empirical Evaluation of Ontology-based Semantic Annotators

Srecko Joksimovic
School of Interactive Arts and Technology,
Simon Fraser University, Canada
sjoksimo@sfu.ca

Jelena Jovanovic
FON-School of Business Administration,
University of Belgrade
jeljov@gmail.com

Dragan Gasevic
School of Computing and Information Systems,
Athabasca University, Canada
dgasevic@acm.org

Amal Zouaq
Department of Mathematics and Computer Science,
Royal Military College of Canada
amal.zouaq@rmc.ca

Zoran Jeremic
Ryerson University, Toronto,
Canada
zoran.jeremic@gmail.com

ABSTRACT

One of the most important prerequisites for achieving the Semantic Web vision is semantic annotation of data/resources. Semantic annotation enriches unstructured and/or semi-structured content with a context that is further linked to the structured domain-specific knowledge. In particular, ontology-based semantic annotators enable the selection of a specific ontology to annotate content. This paper presents results of an empirical study of recent ontology-based annotators, namely Stanbol, KIM, and SDArch. Specifically, we evaluated the robustness of these annotators with respect to specific features of ontology concepts such as the length of concepts' labels and their linguistic categories (e.g., prepositions and conjunctions). Our results show that although significantly correlated according to most of the conducted evaluations, tools still exhibit their unique features that could be a topic of new research.

Categories and Subject Descriptors

I.2.7 [**Natural language processing**]: Text analysis; G.3 [**Probability and statistics**]: Correlation and regression analysis; H.3.4 [**Systems and Software**]: Performance evaluation.

General Terms

Performance, Experimentation

Keywords

Automatic semantic annotation, annotation tools, comparative analysis, empirical study

1. INTRODUCTION

During the last decade, significant efforts have been put in the development of semantic annotation frameworks and tools in order to make the annotation process as easy and as precise as possible [6]. The focus of our research work presented in this paper is on automatic semantic annotation of text-based content (e.g. [3], and [1]). Specifically, the paper targets those annotators that allow for the selection and/or customization of the ontology used in the annotation process and aims at comparing some of the most recent such annotators, namely Apache Stanbol, KIM, and SDArch.

Several past research efforts were focused on the analysis of semantic annotation platforms and tools (e.g., [4], and [5]). In general, researchers provided qualitative comparisons of annotators based on several criteria such as the use of standard formats, ontology support, precision, recall, and f1 measure. However, to our knowledge, the evaluation of recent annotators such as Apache Stanbol and Stanbol KLE has not been performed. Another distinctive feature of this study is its goal to investigate and compare the robustness of contemporary semantic annotators with respect to different, primarily linguistic, features of the ontological concepts (e.g., the length and linguistic structure of concept labels).

To address these specific goals, we have conducted an empirical study with three contemporary semantic annotation tools, namely, KIM[1], Apache Stanbol[2], and SDArch[3]. The data collected in the study was used to perform a detailed analysis and comparison of the examined tools based on the following metrics: the number of the extracted concepts; the ability to process concept labels of various lengths; and the ability to process various linguistic structures recognized in the text.

The following section starts with research questions that drove this research work; it also introduces and describes in detail the study design, materials and procedure. In Section 3, we present and discuss the study results in the context of our research questions. Section 4 concludes the paper with a detailed discussion of the study results and recommendations for future development of the examined annotation tools.

2. METHOD

The study was driven by the following research questions:

RQ1: How do the results of the annotators compare to each other with respect to the number of extracted concepts?

RQ2: Is there a significant difference between the annotators w.r.t. the word length of labels of the concepts recognized in

K-CAP '13, June 23-26, 2013, Banff, Canada.
Copyright 2013 ACM 978-1-45-03-2102-0/13/06 ...$15.00.

[1] http://www.ontotext.com/kim

[2] http://incubator.apache.org/stanbol/

[3] http://www.semanticdoc.org/index.php?action=home

text? We were interested to learn about the capabilities of the studied annotation tools to process multi-words expressions.

RQ3: Is there a significant difference between the annotators w.r.t. the linguistic structure of the concepts (i.e., their labels) recognized in text? We wanted to assess the capabilities of the annotation tools to process various linguistic structures such as labels with punctuations, conjunctions, or prepositions.

In order to answer these research questions (RQs) in a manner that does not affect the internal validity of the study results, we setup the same environment for all the selected annotation tools. Specifically, we assured that all the examined tools use the same ontology, and the same document corpus.

To create the document corpus, we issued a query to ACM Digital Library for each concept from ACM Computing Classification System (CCS). Then, we took the top 100 abstracts of each result set. The document corpus established in this way contained 44,927 abstracts, with average abstract length of 129.40 (SD=70.79) words. For each of the collected abstracts, we also obtained the ACM CSS concepts assigned by the papers' authors. This corpus was stored together with information about each document (including its abstract, URI, title, creation date, and assigned keywords) in an RDF store[4].

Since ACM publications are indexed and categorized using ACM CCS, in this study, we used the ACM CCS ontology[5] for the annotation of our document corpus. Descriptive statistics for the ACM CCS ontology are presented in Table 1.

Table 1. The distribution of the concepts assigned to the corpus' documents by their authors ("Assigned") and all the concepts of the ACM CCS ontology ("ACM-CCS"), based on the word length (WL) of the concepts' labels.

Source	Assigned	ACM-CCS
WL 1	758	423
WL 2	24793	630
WL 3	61851	283
WL 4	23875	100
WL 5	9082	19
WL 6	1168	8
WL 7	86	1
WL 8	0	6
Total	**121613**	**1470**

The next step was tools selection, according to the specified criteria. Specifically, a tool should allow for automated semantic annotation of text-based content and should be configurable with a user-selected ontology. Accordingly, we did not consider tools for named entity recognition as they allow only for the extraction of instances of a predefined set of types (typically, person, organization, date, and the like). We also considered the tool's stage of development, and if it allows for the customization of the annotation process. Another selection criterion was the availability of an appropriate API that provides program-based access to the tool's annotation features. Finally, the tools had to be either open source or free of charge for research purposes. Based on these criteria set, we have chosen the following tools: KIM, Apache Stanbol, and SDArch. Being interested in exploring if KIM's Annotation Cleaner module removes also some domain-specific annotations that could be relevant for the document being annotated, we examined two KIM setups: with (KIM) and without the Annotation Cleaner (KIM_NC).

Furthermore, at the time we considered using Apache Stanbol for the study, its semantic annotation process was based on Taxonomy Linking Engine. In the meantime, this engine was deprecated and replaced by Keyword Linking Engine, which is, according to the Stanbol documentation, more modular and better suited for eventual improvements and extensions. Aiming to assess the effects of the new configuration of Stanbol, i.e., the replacement of the Taxonomy Linking Engine with the Keyword Linking Engine, we included both configurations of the tool in the study. Hereafter, Stanbol stands for Stanbol with Taxonomy Linking Engine while Stanbol_KLE designates Keyword Linking Engine.

Besides incorporating our ontology into the annotation pipeline of each tool, all the experiments were conducted using default settings for each tool.

To perform the annotation and analysis of the collected data, we have developed an application comprising of two main modules. The Annotation module is responsible for collecting data about each abstract stored in the RDF store, parsing the abstract, and calling the methods of the appropriate annotator (via its API) to perform semantic annotation of the abstract. This module produces one RDF file for each tool with different metrics (e.g., number of extracted concepts, length of each abstract, and various other statistics). The Analysis module uses the RDF files generated by the Annotation module in order to compute standard descriptive statistics (for RQ3), including mean values and standard deviation, analysis of variance ANOVA (for RQ1), as well as correlation and regressions analysis (for RQ2).

The collected data were analyzed using the SPSS software. For variables having not normally distributed data, we used parametric tests over log-transformed data. The threshold of $p<0.05$ was chosen to designate the statistically significant level.

3. RESULTS AND DISCUSSION

In this section, we present and discuss the results of our statistical analysis in the context of our research questions.

3.1 RQ1: Differences with respect to the number of extracted concepts

The motive for this research question was to determine the ability of a specific tool to extract certain number of concepts. We also tried to reveal the filtering capabilities of each tool, since higher number of extracted concepts does not necessarily mean better performance. Table 2 contains descriptive statistics of the concepts that each of the studied tools found per analyzed abstract. Values in the table are calculated based on the overall number of extracted concepts. When counting the concepts, we considered each unique URI, regardless of its label[6].

We performed a one-way between subjects ANOVA test with the number of extracted concepts as the dependent variable (Table 2). The test revealed a significant difference between the examined annotators w.r.t. the number of extracted concepts [$F(4, 224680) = 7212.97$, $p < .001$]. Pairwise comparisons,

[4] The entire document corpus is available at http://goo.gl/Mv5tJ

[5] http://goo.gl/VU73H

[6] Although, each concept in ACM-CCS ontology has a unique identifier (i.e. URL), there are cases where labels of concepts are the same [http://goo.gl/soE14].

based on the Tukey post-hoc test revealed significant difference ($p < 0.05$) between each pair of tools.

The Annotation Cleaner in KIM removes a significant number of annotations, which is obvious from the comparison between KIM and KIM_NC (Table 2). Stanbol_KLE produces a significantly higher number of concepts, compared to the previous version of this tool (Stanbol). Further, results show that KIM_NC and SDArch performed in a similar fashion (as confirmed by the Pearson's correlation test: r (44926) = 0.99, p < .001), while Stanbol differed significantly from both of them, in the sense that it generated significantly lower number of concepts. In addition, Stanbol and Stanbol_KLE, although being different versions of the same tool, produced low correlated results (r (44926) = 0.62, p < .001). Furthermore, results produced by KIM were roughly equally correlated with the results of all the other tools.

Table 2. Descriptive statistics (mean value and standard deviation) of the number of concepts per analyzed abstract, assigned by authors (1. row) and discovered by the studied tools (the rest of the table)

Annotator /assigned	Mean (SD)	Min-Max
Assigned	2.71(1.12)	0-16
KIM	4.07 (2.96)	0-44
KIM_NC	11.27 (15.00)	0-103
Stanbol	3.71 (3.39)	0-35
Stanbol_KLE	7.94 (5.12)	0-68
SDArch	10.87 (14.92)	0-101
Overall p value	$p < 0.001^{a\text{-}j}$	

Legend: (statistically significant level at p<0.05) [a]KIM vs. KIM_NC; [b]KIM vs. Stanbol; [c]KIM vs. Stanbol_KLE, [d]KIM vs. SDArch, [e]KIM_NC vs. Stanbol; [f] KIM_NC vs. Stanbol_KLE; [g]KIM_NC vs. SDArch; [h]Stanbol vs. Stanbol_KLE; [i]Stanbol vs. SDArch; [j]Stanbol_KLE vs. SDArch.

The results of this research question revealed significant differences between the tools w.r.t. the number of discovered concepts, in great deal of comparisons. It could be said that only one pair, namely KIM_NC and SDArch, had similar results.

3.2 RQ2: Differences with respect to the length of the labels of concepts recognized in the text

In our dataset, most of the concepts assigned by authors are 2 to 5 words in length (Table 1). We could also observe that the ACM CCS ontology contains concept labels of up to 8 words, and that the majority of concepts have 1 to 5 word lengths. Thus it was important to determine whether annotators could handle the full range of word lengths, or if they were "biased" towards annotating simple words (WL1) only.

To answer this research question, we first report, for each tool, the mean value and standard deviation grouped by word length of the extracted concepts' labels (Table 3). We can observe that KIM_NC and SDArch obtained the highest mean value for the concepts with word length equal to 1. On the other hand, Stanbol and Stanbol_KLE performed better than the other tools for concepts with word length of 5, 6 or 7. Only Stanbol demonstrated the capacity to perform annotation with concepts having word length of 8. In fact, Stanbol discovered concepts for each word length (Table 3; Figure 1).

To further explore this topic, we conducted a series of one way between-subjects ANOVA tests followed by Tukey post-hoc tests (for Pairwise Comparison of the tools); each test used word lengths as its dependent variable. The ANOVA test revealed a significant difference among the annotators w.r.t. the number of extracted concepts with (labels of) word lengths 1-5. Subsequent pairwise comparisons revealed significant differences ($p < 0.001$) in the number of discovered concepts of word length 1 and 2, between each pair of the examined tools. However, this was not the case with other word lengths. Only Stanbol and Stanbol_KLE discovered concepts of higher word lengths (6, 7, 8); however, the post-hoc tests did not reveal any significant difference between these two tools for word lengths 6-8.

Table 3. Descriptive statistics grouped by word length of the extracted concepts' labels

Tool	KIM	KIM_ NC	Stanbol	Stanbol _KLE	SDArch	Overall p value	F(4, 224680)
WL1	3.69 (2.72)	10.80 (14.86)	3.16 (2.98)	6.49 (4.35)	10.37 14.77	$p < .001^{a\text{-}j}$	7150.45
WL2	0.35 (0.65)	0.43 (0.90)	0.19 (0.54)	0.75 (1.16)	0.46 (0.94)	$p < .001^{a\text{-}j}$	2744.76
WL3	0.03 (0.18)	0.03 (0.22)	0.2 (0.55)	0.65 (0.99)	0.04 (0.23)	$p < .001^{b\text{-}f,\ h\text{-}j}$	13303.32
WL4	0.002 (0.05)	0.002 (0.05)	0.09 (0.33)	0.04 (0.22)	0.003 (0.06)	$p < .001^{b\text{-}c,\ e\text{-}f,\ h\text{-}j}$	2112.80
WL5	4 E-5 (0.007)	4 E-5 (0.007)	0.04 (0.24)	0.003 (0.05)	7 E-5 0.008	$p < .001^{b\text{-}c,\ e\text{-}f,\ h\text{-}j}$	1818.38
WL6	0 (0)	0 (0)	0.005 (0.08)	0.0002 (0.01)	0 (0)	-	-
WL7	0 (0)	0 (0)	0.005 (0.07)	2 E-5 0.005	0 (0)	-	-
WL8	0 (0)	0 (0)	0.005 (0.1)	0 (0)	0 (0)	-	-

Legend: (statistically significant level at p<0.05) [a]KIM vs. KIM_NC; [b]KIM vs. Stanbol; [c]KIM vs. Stanbol_KLE, [d]KIM vs. SDArch, [e]KIM_NC vs. Stanbol; [f] KIM_NC vs. Stanbol_KLE; [g]KIM_NC vs. SDArch; [h]Stanbol vs. Stanbol_KLE; [i]Stanbol vs. SDArch; [j]Stanbol_KLE vs. SDArch

Figure 1 represents the percentage of the overall number of discovered concepts for each considered word length. It allows for a seamless comparison of the word-length-based distribution of: i) the concepts extracted by each examined tool, ii) the concepts assigned by authors, and iii) the overall concepts in the ACM CCS ontology. This figure clearly indicates a "bias" of the annotators towards concepts with labels of word length 1, whereas the ACM CCS ontology itself and the concepts from ACM CCS assigned by authors are mainly distributed between concepts with labels of word length 2-5. This is likely an indicator of a need for the use of deeper NLP techniques in semantic annotators.

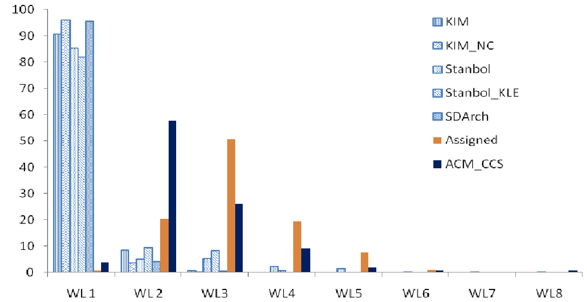

Figure 1. Percentage of the overall number of discovered concepts for each considered word length

We also measured, using Pearson's correlation test, the strength of the association between the number of concepts discovered by each annotation tool and word length of the concepts' labels. The obtained results confirmed that concepts of word length 1 were the most extracted ones by all annotators. The number of concepts discovered using KIM, KIM_NC and SDArch were significantly correlated with word length 1 to 4. Linear regression analysis confirmed these correlation results[7].

These findings further confirm our findings related to RQ1 (Sect. 3.1). In most comparisons, KIM_NC and SDArch performed in a similar fashion, while the results obtained by using KIM differ significantly compared to these two tools. Thus, we were able to conclude that overall, KIM applies filtering better than SDArch, while differences between KIM and KIM_NC are in line with our expectations. Results also reveal that Stanbol, although being an older version of the tool, performs better filtering than Stanbol_KLE.

3.3 RQ3: Differences with respect to the concepts' linguistic structures

Finally, we wanted to compare the type of linguistic structures handled by the considered semantic annotators. The objective was to explore how the annotators handle these structures and whether their presence influences the extraction process.

We first identified the following linguistic constructs in the ACM CCS ontology: simple words (e.g., "Data"); multi-words expressions (e.g., "computer systems organization"); hyphenated compounds (e.g., "special-purpose and application-based systems"); punctuations, as expressions containing a punctuation sign (e.g., "lists, stacks, and queues"); conjunctions, as expressions containing the co-ordinate conjunctions "and" and "or" (e.g., "coding and information theory"); and prepositions, as expressions containing a preposition, mainly "of' (e.g., "theory of computation"). Note that the same label might be assigned to multiple categories such as "special-purpose and application-based systems" which is categorized both as a hyphenated compound and a conjunction.

Summary of the number of concepts of the ACM CCS ontology pertaining to each linguistic category as well as descriptive statistics about the annotations manually assigned by the authors, and the concepts extracted by the examined annotators, are available at [2]. We were able to conclude that all the annotators were able to annotate each linguistic category. However, simple words are significantly more present among recognized concepts, than the concepts assigned by authors (Sect. 3.2, and Figure 1). On the other hand, in the case of other categories (multi-word expressions, hyphenated compound, conjunctions, punctuations and prepositions), authors assigned significantly higher number of concepts than tools were able to discover. We also found that complex concept labels (prepositions, compounds, conjunctions, etc.) have often all their simple words annotated separately in text, if those simple words exist as concepts in the ontology. For example, in concept "*Parallel and vector implementations*", "and" should be interpreted as having a distributive meaning: "*Parallel implementation*" and "*Vector implementation*". According to our observations, the annotators cannot detect such structures. Still, based on the randomly selected samples, we noticed that Stanbol_KLE was the most effective annotator in handling complex linguistic expressions such as "*Biology and Genetics"* or "*Simplification of expressions"*. Some annotators such as KIM, KIM_NC and SDArch indicate some adjectival modifiers as annotations (e.g. "Distributed" instead of "distributed systems"), which should be avoided.

4. CONCLUSION AND FUTURE WORK

Due to the space limit, we were not able to present and discuss more thoroughly the random samples we extracted in order to examine the capacity of the selected annotators to handle different linguistic structures. However, we find it important to stress the fact that simple words are the most represented category among all analyzed tools. Besides that, some annotators (KIM, KIM_NC, SDArch) often tag both multi-word expressions and their components (e.g. simulation languages, simulation, languages), which might explain the huge number of annotations. We also observed that annotators generally extract exact concepts labels; some lemmatization for handling plural/singular words is only observed in the case of Stanbol and Stanbol_KLE. Singular/plural forms in the ACM CCS ontology failed to be discovered by KIM, KIM_NC and SDArch and we are intending to explore this further. Exact labels are also discovered in the punctuation or hyphenated compounds categories, which might be a problem, as it is likely that some punctuation signs might differ in texts.

Finally, we need to acknowledge that this study relied on the basic configuration of the examined annotators. This was done on purpose to measure the effectiveness of the configurations that are likely to be adopted by a vast majority of users. In fact, although it is possible to tailor each tool to annotate a given domain better, this is not a trivial task. Moreover, the Semantic Web uptake necessitates tools that are made for end users (in this case, developer) who are not ontology/knowledge engineers. As such, these semantic annotators seem not ready for a straightforward and successful adoption by end users, although Stanbol with Keyword Linking Engine (Stanbol_KLE) seems to be a good step in the right direction.

In future work, we intend to assess the capabilities of each tool to annotate texts with concepts from different hierarchy levels in the ontology and to enrich our study with a more detailed qualitative analysis, primarily annotation pipeline analysis.

5. REFERENCES

[1] Andrews, P., Zaihrayeu, I., & Pane, J. (2012). A Classification of Semantic Annotation Systems. *Semantic Web Journal*, in press.

[2] *http://bit.ly/11aDjGG*. (n.d.).

[3] Maynard, D. (2008). Benchmarking Textual Annotation Tools for the Semantic Web. *Proc. of the 6h Int'l Language Resources and Evaluation (LREC'08)*, (pp. 20-25).

[4] Reeve, L., & Han, H. (2005). Survey of Semantic Annotation Platforms. *Proceedings of the 2005 ACM Symposium on Applied Computing*.

[5] Uren, V., Cimiano, P., Iria, J., Handschuh, S., Vargas-Vera, M., Motta, E., & Ciravegna, S. (2005). Semantic annotation for knowledge management: Requirements and a survey of the state of the art . *Journal of Web Semantics*. Van Harmelen, F. (2000). The Semantic Web: the Roles of XML. *IEEE Internet Computing, vol. 15, No. 3*, 63-74.

[7] The results of these analyses are available at: http://bit.ly/12JIJy2