A Comparison of NER Tools w.r.t. a Domain-Specific Vocabulary

Timm Heuss
University of Plymouth
Plymouth, United Kingdom
Timm.Heuss@plymouth.ac.uk

Timm Heuss
University of Applied Sciences
Darmstadt, Germany
Timm.Heuss@h-da.de

Bernhard Humm
University of Applied Sciences
Darmstadt, Germany
Bernhard.Humm@h-da.de

Christian Henninger
University of Applied Sciences
Darmstadt, Germany
Christian.Henninger@stud.h-da.de

Thomas Rippl
University of Applied Sciences
Darmstadt, Germany
Thomas.Rippl@stud.h-da.de

ABSTRACT

In this paper we compare several state-of-the-art Linked Data Knowledge Extraction tools, with regard to their ability to recognise entities of a controlled, domain-specific vocabulary. This includes tools that offer APIs as a Service, locally installed platforms as well as an UIMA-based approach as reference. We evaluate under realistic conditions, with natural language source texts from keywording experts of the Städel Museum Frankfurt. The goal is to find first hints which tool approach or strategy is more convincing in case of a domain specific tagging/annotation, towards a working solution that is demanded by GLAMs world-wide.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*; H.2.8 [**Information Systems**]: Database Applications—*Data mining*

General Terms

Measurement

Keywords

Named Entiy Recognition, Linked Data, Domain-Specific Vocabulary

1. INTRODUCTION

Keywording artworks, exhibits and other objects in galleries, libraries, archives and museums (GLAMs) with a common vocabulary has been subject of the catalogisation efforts of many years now [15]. In this context, a number of authority files have been developed, like the VIAF, or the GND, meeting the very specific needs of libraries or museums and fostering the exchange of catalog meta data. Today, many of these authority files are available as Linked Data, a fact that simplifies the reuse and exchange of data. However, the major and most important part of the catalogization efforts, the unique identification of entities, remains to be done at a low level of automatic assistance.

SEM'14 September 04 - 05 2014, Leipzig, AA, Germany

http://dx.doi.org/10.1145/2660517.2660520.

2. MOTIVATION

In the last years, a number of tools have emerged, combining both Natural Language Processing (NLP) and Linked Data capabilities to conduct a fully- or semi-automated Named Entity Recognition (NER) of Linked Data entities [2]. Recent work of Aldo Gangemi compares the performances of such tools in their recognition of general purpose entities in a New York Times news article [2]. In the cultural heritage context, [3] have compared some tools in their ability to link to the Linked Data cloud.

However, instead of dealing with commonly texts, vocabulary and entities, keywording efforts in GLAMs are often in the context of a specific domain, with specific texts, vocabulary and entities, sometimes even with a specific use of language. The choice for a certain authority file that is used for the keywording has impacts on the entire process, as it not only defines the entities, but also the level of atomicity in which these entities are disambiguated. Exactly this choice is made by keywording and domain experts, based on how well the expressions of an authority file fit to the specific matter - and not how well they are connected to the Linked Data cloud.

In our case, for example, the Städel museum in Frankfurt seeks to keyword its exhibits, based on the associated description texts. For this use case, the museum has selected the public authority file Iconclass [11], as it is the most fitting vocabulary to the demands of the local keywording experts. And because the Städel museum has about 100.000 exhibits, an automated solution or even a semi-automated assistance is very preferable.

The question is: given the many tools and approaches available on the one hand, and the domain-specific use case

on the other, what is the best approach for the domain specific NER? Thus, subject of this paper is a comparison of several NER approaches in a controlled, domain-specific vocabulary, in this case Iconclass, available as Linked Data[1]. Thereby, three different kinds of approaches are taken into consideration: (1) Software as A Service, that need to be queried via specific APIs, (2) locally installed NER platforms that can be configured, and a (3) pipeline-based, general-purpose NLP tools, programmed and configured for the very specific scenario.

3. EVALUATION PROCEDURE

This specific evaluation involves three kinds of inputs: (1) a controlled domain-specific vocabulary (here: Iconclass), (2) a set of domain-specific texts in different languages (here: German and English), and (3) a golden standard, containing the to-be-detected named entities of the controlled domain-specific vocabulary within each text, created by a human expert. The primary evaluation criteria is the quality of the NER, measured in precision, recall, and F1 measure, compared to the golden standard. We also compare implementation efforts and the requirements needed for the integration and use.

3.1 About Iconclass

Iconclass - available as Linked Data[1] - is a library classification for art created by Henri van de Waal [10] and has been selected by the Städel museum to be used for keywording certain attributes of exhibits. Today, the controlled vocabulary is one of the biggest and widespread classification system for art, consisting of ten categories with 28,000 hierarchically ordered concepts [11]. Each concept consists of a unique identifier, links to *broader* and *narrower* concepts, as well as labels (*prefLabel*) and keywords (*subjects*) in multiple languages (English, German, Italian, French).

In the following, the RDF of a single Iconclass concept, *11F244*, is excerpted for the languages German and English - using the usual namespaces for *skos* and *dc*, as well as using namespace *ic* for *http://iconclass.org/*.

```
ic:11F244 a skos:Concept .
ic:11F244 skos:prefLabel
                  "die thronende Maria"@de .
ic:11F244 skos:prefLabel
                    "Mary enthroned"@en .
ic:11F244 dc:subject "Maria (Jungfrau)"@de .
ic:11F244 dc:subject "Religion"@de .
ic:11F244 dc:subject "Thron"@de .
ic:11F244 dc:subject "sitzen"@de .
ic:11F244 dc:subject "uebernatuerlich"@de .
ic:11F244 dc:subject "Mary (Virgin)"@en .
ic:11F244 dc:subject "religion"@en .
ic:11F244 dc:subject "sitting"@en .
ic:11F244 dc:subject "supernatural"@en .
ic:11F244 dc:subject "throne"@en .
ic:11F244 skos:broader ic:11F2 .
ic:11F244 skos:broader ic:11F24 .
ic:11F244 skos:narrower ic:11F244%28%2B0%29.
ic:11F244 skos:narrower ic:11F244%28%2B1%29.
```

3.2 Scenario

In the context of this paper, Iconclass subjects represent the entities that need to be detected by the NER tools. Whenever tools do not produce those subjects natively according to Iconclass (and can not be configured to do so), the specific entities detected by the tools are mapped to Iconclass subjects, based on their string labels. We consider this mapping to be a valid move, as subjects in Iconclass are also only present as simple strings.

The input texts from the Städel museum are written in German in different styles. They contain about 300 words and the corresponding, human-created golden standard matches against 55 Iconclass labels or subjects. The following line shows a short sample for the German original text [17], expected keywords are indicated by curly brackets:

```
{Maria} in {Vorderansicht}, [...] auf einem
    {Throne} {sitzend}, [...]
```

In this sample, the keywords *Maria*, *Throne* and *sitzen* are subjects of the previously introduced concept *11F244*.

We also created machine-translated English versions of the German source texts, which are used in every tool that does not support German.

Besides the actual NER performance, measured in precision, recall and F1 compared to the golden standard, details in the output format of the tools are essential for further processing. Thus, this comparison also contains detailed information about the specific output of each individual tool.

4. NER TOOLS

The tool selection in this comparison bases on the work of [2] and includes the best Named Entity Recognition (NER) performers in their test, archiving a F1 score of 48% or better. This involves several As-A-Service-solutions, as well two locally installed platforms. For a third category of tools - NER pipelines built by hand - two well-known frameworks come into question: GATE[2] and UIMA[3]. Since both approaches offer similar possibilities and features, we only implemented an UIMA pipeline.

4.1 As-A-Service Tools

4.1.1 AlchemyAPI

AlchemyAPI is a web service that analyses unstructured content (news articles, blog posts, e-mail, etc.) as text or web-based content and identifies named entities (people, locations, companies, etc.), facts and relations, topic keywords, text sentiment, news and blog article authors, taxonomy classifications, scraping structured data, and more. It supports English, German and 6 more languages. The free API-Key includes 1,000 calls a day. [9]

4.1.2 CiceroLite

CiceroLite is developed by Language Computer Corporation and part of the Cicero On-Demand Server. This is a commercial software, for our review we used the demo API available on for testing purposes. It supports English, Modern Standard Arabic, Mandarin Chinese, Japanese Spanish, German and Dutch texts for named entity recognition. The demo API requires German input texts. The API supports REST and thus can be used from within most programming

[1] http://www.iconclass.org/help/lod (accessed 2014-07-25).

[2] https://gate.ac.uk/projects.html (accessed 2014-07-26).

[3] https://uima.apache.org/ (accessed 2014-07-26).

languages. The result[4] also contains information about the topics of the text and a HTML result viewer that shows the annotated result. [12]

4.1.3 FOX

FOX is developed by the Leipzig University and uses of the diversity of NLP algorithms to extract RDF triples. In its current version, it integrates and merges the results of Named Entity Recognition, Keyword Extraction and Relation Extraction tools. It can be used programmatically or as a web application. [4]

4.1.4 FRED

FRED is a free text annotation service developed by the university of Bologna. The service is free to use and available via REST. The web interface also shows a graph of the structure of the input text. The only supported language is English at the moment. [1]

4.1.5 NERD

The NERD framework combines multiple extracting engines to achieve its results [16]. These engines are: AlchemyAPI, dataTXT, DBpedia Spotlight, Lupedia, OpenCalais Saplo, SemiTags, TextRazor, THD, Wikimeta, Yahoo! Content Analysis and Zemanta [6]. To make the result more readable and to avoid entities to appear multiple times in the result, the results are mapped to the NERD ontology. The number of API calls per day is restricted to 500.

4.1.6 Open Calais

Open Calais is a service by Thomas Reuter to extract information from texts which was released in 2008. The software was developed and used mainly to tag articles by keywords such as blogposts. The API is free for up to 50000 calls a day. It supports texts in english, french and spanish. [5]

4.1.7 Wikimeta

Wikimeta is a API to semantically annotate texts. The basic version has a limit of 100 calls per day, several plans including a dedicated server are available for commercial users. Any user gets a unique API-Key as well as access to a configuration site. This site appeared broken during the evaluation. It supports english, spanish and french texts. A perl and Java example is available to show how to access the service, due to the fact that it is based on a REST service most programming languages are supported. [13]

4.1.8 Zemanta

Zemanta is a WordPress plugin for bloggers that analyzes content and returns relevant metadata (tag, entities, categories) as well as content enhancements such as related articles and image links. The functionality of the plugin is also provided as an API. The API is available for C#, Java, JavaScript, Perl, PHP, Python and Ruby. In order to run the API requires an API-Key, which is available for free and includes 1000 calls per day.

[4]During our evaluation, the results of Cicero Lite slightly varied from call to call, even when fired consecutively. Due to the lack of source code to evaluate this behaviour, most likely machine learning algorithms are responsible for this behaviour. Note that the results presented are the best we encountered during multiple evaluation runs.

4.2 Locally Installed Platforms

4.2.1 AIDA

AIDA is a named entity disambiguation system developed by Max Planck Institute for Informatics in Saarbrücken, Germany. It identifies mentions of named entities (e.g. persons or locations) in English language text and links them to a unique identifier. The mentions identified are registered in the Wikipedia-derived YAGO2 knowledge base. AIDA is written in Java and licensed under CC BY-NC-SA 3.0.

4.2.2 Apache Stanbol

Apache Stanbol is a collection of components for semantic content management developed under the APL.The enhancer feature annotates text sources according to various available Enhancement Chains. These chains can be configured and extended. Own ontologies can be imported and used for entity recognition by the entity hub. The software also provides extensions for rules and content management systems. The default installation supports 7 languages. [8]

Detection in Stanbol depends on the chosen Enhancement Chain. [7]. In this evaluation, we used two different configurations of such Enhancement Chains. The first uses the default "dbpedia-disambiguation" chain, and is in the result pages refered as *Stanbol (default)*. However, the actual benefit of Stanbol is the fact that custom vocabularies can be installed. So we created a second chain, called *Stanbol (Iconclass)*, based on the components of the default one. To recognise the terms of our controlled, domain-specific vocabulary, the dataset was indexed and imported in the so called Entityhub. We created a new Linker, specifying the field which is used to match against and other attributes. The new Linker was placed before the dbpedia linker in the toolchain.

4.3 Customized Pipeline

Customized Pipeline is programmed exactly for the detection scenario in this paper, and bases on UIMA-NLP-components. Besides the usual NLP tasks, like segmentation, parsing, lemmatization, and n-gram-creation, the Customized Pipeline contains a Iconclass subject matcher, based on simple string-matching and detection heuristics. The entire pipeline is straight forward and lacks advanced features, like a spell check mechanism or a context recognition.

5. TEST-SETUP

For the evaluation in this paper, we developed a simple test framework that submits the German and English texts to each individual NER tool (depending on its language support), parses the results, filters out non-Iconclass-words, and compares them to the expected results in the golden standard. Thereby, strings are simply matched against the golden standard, without regard of their case; and the counters for true positives (tp), false positives (fp) or false negatives (fn) are increased accordingly. tp is the number of strings that were expected and successfully identified, fp is the number of strings that were identified but not expected and fn is the number of strings that were expected but not identified. After the evaluation is completed for a tool, the counters are used to calculate precision, recall and F1 as usual [14].

6. EVALUATION RESULTS

Tool Name	Requirements	Setup efforts
AlchemyAPI	AlchemyAPI SDK and API-Key	Low - Include the SDK to the project. Code examples are available.
CiceroLite	No requirements	Low - No setup required. API call is realized via HTTP-Post request. Example request is available.
FOX	No requirements	Low - No setup required. API call is realized via HTTP-Post request. Example request is available.
FRED	No requirements	Low - No setup required. API call is realized via HTTP-Post request. Example request is available.
NERD	API-Key	Low - No setup required. API call is realized via HTTP-Post request. Example request is available.
Open Calais	API-Key	Low - No setup required. API call is realized via HTTP-Post request. Example request is available.
Wikimeta	API-Key	Low - No setup required. API call is realized via HTTP-Post request. Example request is available.
Zemanta	Zemanta API and API-Key	Low - Include the API to the project. Code examples are available.
AIDA	Maven, postgres database with an entity repository (e.g. YAGO2) and a clone of the AIDA repository	Medium - Setup the postgres database with the entity repository. Configure and build AIDA library with maven. Code examples are available.
Apache Stanbol	Maven, Skills to configure Stanbol	Medium - Maven built from sources. In order to handle a custom dictionary another Jar-file must be built with Maven, configured and the resulting indexed Store must be imported.
Customized Pipeline	Maven Repository, Language Modells, UIMA skills, linguistic basics	High - Find required NLP components, combine them in a specific pipeline, solve language specific issues, select proper models, write Iconclass importer

Table 1: Outline of the efforts and requirements to use the tools in this evaluation.

Tool name	Output Format	Entities identified as		Contains information about the sentence structure
		String / Typed String	as Identifier of public data	
AlchemyAPI	RDF, XML, JSON	+ / +	DBpedia, Freebase, YAGO	-
CiceroLite	JSON, RDF, HTML viewer, XML	+ / +	DBpedia	+
FOX	RDF/XML, RDF/JSON, JSON-LD, N3, N-Triple, TURTLE	+ / +	DBpedia	String position only
FRED	Graphical, RDF, DAG	+ / +	DBpedia, WordNet	+
NERD	JSON	+ / +	DBpedia	String position only
Open Calais	XML/RDF, Text/Simple, Text/Micro-formats, JSON, Text/N3	+ / +	DBpedia, Freebase, Reuters.com, GeoNames, Shopping.com, LinkedMDB	
Wikimeta	JSON, XML	+ / +	DBpedia	+
Zemanta	RDF/XML, JSON	+ / -	-	-
AIDA	Own format	+ / +	YAGO	String position only
Apache Stanbol	JSON-LD, RDF/XML, RDF/JSON, Turtle, N-Triples	+ / +	DBpedia	
Customized pipeline	UIMA-specific (CAS)	+ / +	-	+

Table 2: Details on the information contained in the responses of the tools.

Tool Name	English			German		
	Precision	Recall	F1	Precision	Recall	F1
AlchemyAPI	0.53	0.20	0.29	1.00	0.15	0.26
CiceroLite	0.82	0.20	0.32	1.00	0.07	0.14
FOX	0.83	0.11	0.20	-	-	-
FRED	0.30	0.27	0.28	-	-	-
NERD	0.67	0.44	0.53	0.63	0.09	0.16
Open Calais	0.25	0.02	0.04	-	-	-
Wikimeta	0.43	0.07	0.12	-	-	-
Zemanta	0.50	0.02	0.04	-	-	-
AIDA	0.71	0.11	0.19	-	-	-
Stanbol (default)	0.13	0.02	0.03	0.50	0.07	0.12
Stanbol (IconClass)	0.64	0.50	0.56	0.64	0.50	0.56
Customized Pipeline	0.51	0.78	0.61	0.62	0.52	0.57
Average	0.53	0.23	0.27	0.75	0.26	0.33
Standard derivation	0.22	0.24	0.21	0.23	0.23	0.22

Table 3: Evaluation results, per tool and language.

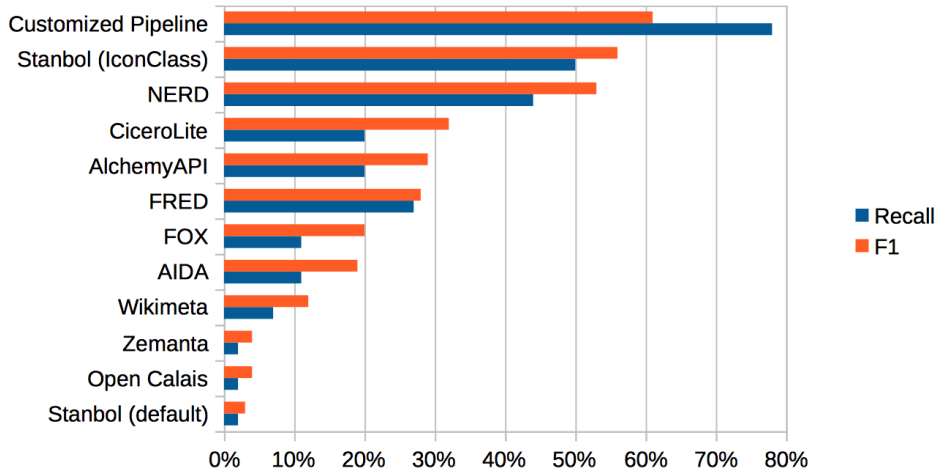

Figure 1: Visualized F1 and recall measures, per tool, English source text.

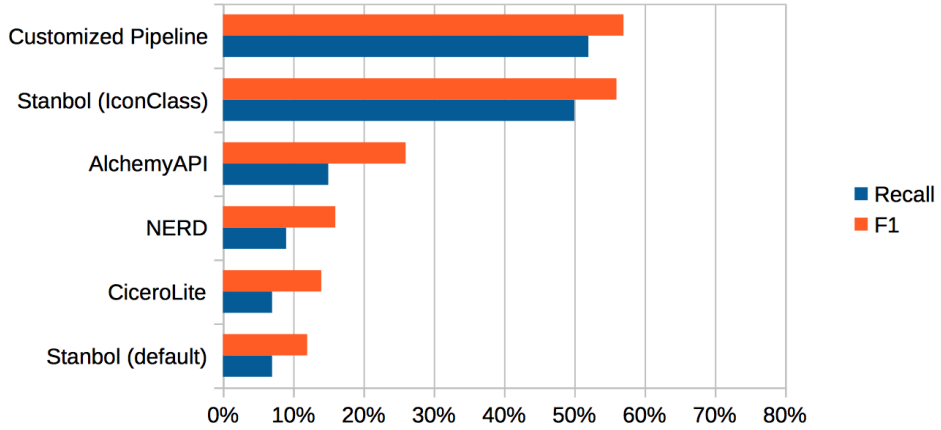

Figure 2: Visualized F1 and recall measures, per tool, German source text.

Tool Name	this evaluation (domain-specific)		Gangemi [2] (general-purpose)		F1	Hooland [3]		F1
	Recall	F1	Recall	F1	Difference	Recall	F1	Difference
AIDA	0.11	0.19	0.57	0.73	-74%	-	-	-
AlchemyAPI	0.20	0.29	0.57	0.73	-60%	0.30	0.41	-29%
Apache Stanbol	0.50	0.56	0.43	0.48	+17%	-	-	-
CiceroLite	0.20	0.32	0.79	0.79	-59%	-	-	-
FOX	0.11	0.20	0.50	0.64	-69%	-	-	-
FRED	0.27	0.28	0.57	0.64	-56%	-	-	-
NERD	0.44	0.53	0.79	0.76	-30%	-	-	-
Open Calais	0.02	0.04	0.50	0.58	-93%	-	-	-
Wikimeta	0.07	0.12	0.71	0.71	-83%	-	-	-
Zemanta	0.02	0.04	0.79	0.71	-94%	0.44	0.57	-93%
Average	0.19	0.26	0.62	0.68	-60%	0.37	0.49	-52%
Standard derivation	0.17	0.18	0.14	0.09	33%	0.10	0.11	45%

Table 4: Comparison of the NER performances in this evaluation (specific input texts, specific vocabulary) to the evaluation of Gangemi [2] (common input texts, common vocabulary) and Hooland et al. [3] (specific input texts, common vocabulary).

7. OBSERVATIONS

A first observation is the fact that the domain-specific scenario in this paper has significant impacts on the overall detection performance, compared to a general purpose scenarios. Table 4 compares the English F1 and recall scores with Gangemi [2] and Hooland et al. [3]

The F1 performances dropped by an average of 60%, having in addition a very high standard deviation of 33%. Roughly said, on average, every tool archived only a third of its F1 scores in common scenarios. Only one single tool, Apache Stanbol, performed better in this specific scenario than in a common scenario. This is due to its relatively low performance in the common scenario in the first place, but also thanks to the fact that Stanbol can be customized and thus fine-tuned for the detection of a controlled domain-specific vocabulary.

As Figure 1 shows, the actual detection rates are low: the best solution in this evaluation archives a F1 of 61%, while the best in [2], for example, scored 85%. Also, the very high standard derivations for recall, precision and F1 are remarkable. So not only the performances are lower than in [2] or [3], the results are also distributed in a larger range. So, for example, the recall scores in this evaluation range from about 78% to only 2%. This means in the worst case, some tools do not detect 98% of the domain-specific entities within the text.

When it comes to the several kinds of tools that are involved, this evaluation shows tendencies that if a custom vocabulary like Iconclass can be loaded (configured or programmed) into the tool, it significantly improves the NER. Consider the remarkable difference of Stanbol (default), loaded with general purpose vocabulary, and Stanbol (Iconclass), loaded with controlled, domain-specific vocabulary. This is also true for the pipeline that was specifically programmed for this scenario. Nevertheless, it archived a F1 of only 61% - even it is still the best solution in this evaluation, this detection rate is too low to be directly used in practice. We think this is to due the use of domain-specific wording in the source texts, as well as tricky and unusual compounds, like *Brocatstoff*. Also, because it is straight-forward and lacks advanced features, there is still room for improvements, especially in regard of the many false positives that caused a relatively bad precision score. This is due to the fact there is currently no context recognition.

Another observation is for German texts: the medium effort Stanbol (Iconclass) almost performs as well as the high-effort, Customized Pipeline. This effect can approximately be observed in the English samples, too, tough the recall of the Customized Pipeline is significantly higher.

The performance of NERD is also remarkable in the context of the implementation effort: even if it is a low-effort service, it nearly archives the scores of the medium-effort Stanbol (Iconclass), with a F1 difference of about 5%. To our knowledge, NERD does not include Iconclass, but combines several different NER tools under the hood.

Majority of tools only supported English input texts. In addition, even if German was the original language of the source texts and could therefore be considered to be of highest quality, and the English versions are the result of a machine translation, tools that supported both languages usually performed better in English. Again, this can be explained with the complex wording and compounds.

Please note that we only evaluated the NER performance for our specific scenario. Additionally, some tools have very useful features, besides NER, that could not be respected in this paper.

Also note that F1 might not be the most relevant measure to evaluate NER tools, especially when a human expert is involved in the keywording process. If a human can confirm or reject keyword findings, we would consider recall the more suitable measure, as it indicates the keyword yield and does not include statements to the correctness of findings.

By considering the exact keywords returned by each of the three best NER performers in this evaluation, Stanbol (Iconclass), Customized Pipeline and NERD, it becomes obvious how the results are distributed. As Figure 3 shows, in the English text, 37% of the keywords in the golden standard are identified by Stanbol (Iconclass) and the Customized Pipeline simultaneously. So majority of the results of Stanbol's 46% and the Customized Pipeline's 51% are identical. This looks different when including NERD: NERD could contribute about half of its keywords as unique findings, e.g.

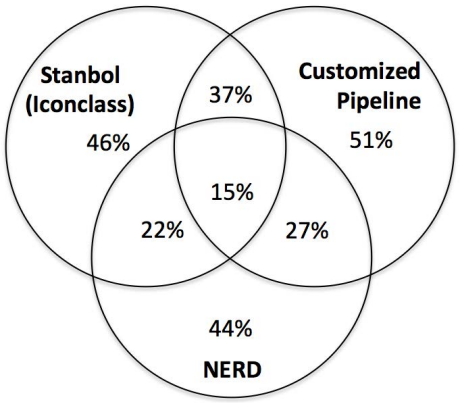

Figure 3: Overlapping of NER results of the three best performers (English source text), 100% = all keywords in the golden standard.

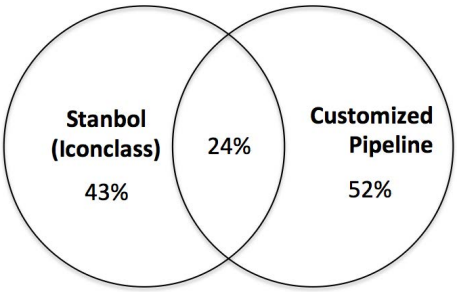

Figure 4: Overlapping of NER results of the two best performers (German source text), 100% = all keywords in the golden standard.

when working together with Stanbol. Just by combining the low- and the medium-effort approaches, the high-effort solution could be outperformed.

In contrast, the German results (Figure 4) show that about a half of the results are distinct between Stanbol and the Customized Pipeline.

8. CONCLUSION

In this paper, we evaluated and compared a number of approaches for the automatic Named Entity Recognition, in the specific context of a museum, in regard to a controlled, domain-specific vocabulary.

It was shown that this domain-specification has significant impacts to the archived detection performance, leading to F1 droppings of 60% ± another 30%, compared to common detection scenarios. Only three tools score a F1 of $> 50\%$.

The results show first evidence that it is generally preferable if a given controlled, domain-specific vocabulary can be loaded into the respective NER tool - a feature that most solutions included in this evaluation do not offer.

However, this involves at least medium or high engineering efforts, even though Linked Data-based approaches simplify this process. For the cases this is not affordable, a combination of several different general-purpose detection tools seems to be promising, like done by NERD.

Detailed review of the tool responses showed that although not a single tool satisfies real-world demands in entity detection individually, in this case, a combination of the three mentioned tool chains could be a succeeding strategy.

9. FUTURE WORK

We will extend the evaluation and experiment with different kinds of combinations of Stanbol, the Customized Pipeline and NERD.

The pipeline does not yet produce unique, disambiguated Iconclass concepts, but the prerequisite of it, Iconclass subjects and labels. So in addition to optimising the NER for a better recall, we will put our efforts in heuristics to find those concepts. Again, in addition to the specific structure of Iconclass, world knowledge as Linked Data, e.g. from WordNet, will support us in this endeavour.

It can be assumed that we do not need a single, complete, fully automatic NER tool, but a pipeline consisting of several steps, whereas a first step is the automatic detection of possible keywords, followed by a number of components for domain and dataset specific rules, which disambiguate, add or remove certain findings, and a last step with a human expert, confirming the results.

10. ACKNOWLEDGEMENTS

We would like to thank Kathleen Benecke, Esther Woldemariam, Gabi Schulte-Lünzum, Ulrike Fladerer and Saskia Lorenz from the Städel Museum for their natural language input texts which enabled us to test in a realistic scenario.

Thanks to Marcus Pohl for his help in developing and evaluating the UIMA-based pipeline.

This work has been funded by the Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz (LOEWE) of the state Hesse, under grant HA 321/12-11. We thank the University- and State Library Darmstadt, the Städel Museum Frankfurt, the media transfer AG, the Software AG, the House of IT and the nterra GmbH for their contribution in this research project.

11. REFERENCES

[1] F. Draicchio, A. Gangemi, V. Presutti, and A. G. Nuzzolese. FRED: From Natural Language Text to RDF and OWL in One Click. In *ESWC (Satellite Events)*, pages 263–267, 2013.

[2] A. Gangemi. A Comparison of Knowledge Extraction Tools for the Semantic Web. In P. Cimiano, O. Corcho, V. Presutti, L. Hollink, and S. Rudolph, editors, *The Semantic Web: Semantics and Big Data*, number 7882 in Lecture Notes in Computer Science, pages 351–366. Springer Berlin Heidelberg, Jan. 2013.

[3] S. v. Hooland, M. D. Wilde, R. Verborgh, T. Steiner, and R. V. d. Walle. Exploring entity recognition and disambiguation for cultural heritage collections. *Literary and Linguistic Computing*, page fqt067, Nov. 2013.

[4] http://aksw.org/Projects/FOX.html. FOX - Federated knOwledge eXtraction Framework, 2 2014. Last access 2014-02-25.

[5] http://en.wikipedia.org/wiki/Calais_(Reuters_Product). Calais (Reuters product) - Wikipedia, the free encyclopedia, 5 2014. Last access 2014-07-28.

[6] http://nerd.eurecom.fr/documentation#extractors. NERD: Named Entity Recognition and Disambiguation, 2014. Last access 2014-07-28.

[7] https://stanbol.apache.org/docs/trunk/customvocabulary.html. Apache Stanbol - Working with Custom Vocabularies, 2014. Last access 2014-06-15.

[8] https://stanbol.apache.org/index.html. Apache Stanbol - Welcome to Apache Stanbol!, 2010. Last access 2014-06-15.

[9] http://www.alchemyapi.com/products/products overview/. Products Overview | AlchemyAPI, 7 2014. Last access 2014-07-28.

[10] http://www.iconclass.nl/about-iconclass/history-of iconclass. History of Iconclass — Iconclass, 2012. Last access 2014-07-26.

[11] http://www.iconclass.nl/about-iconclass/what-is iconclass. What is Iconclass? — Iconclass, 6 2014. Last access 2014-06-13.

[12] http://www.languagecomputer.com/ciceroserver. Language Computer - Cicero On-Demand API, 6 2014. Last access 2014-06-15.

[13] http://www.wikimeta.com/faq.html. Wikimeta |The text-mining and semantic annotation architecture, 2 2014. Last access 2014-02-14.

[14] C. D. Manning, P. Raghavan, and H. Schütze. *An Introduction to Information Retrieval*. Number c. Cambridge University Press, New York, United States of America, 2009.

[15] J. Neubert and K. Tochtermann. Linked Library Data: Offering a Backbone for the Semantic Web. In D. Lukose, A. R. Ahmad, and A. Suliman, editors, *Knowledge Technology*, number 295 in Communications in Computer and Information Science, pages 37–45. Springer Berlin Heidelberg, Jan. 2012.

[16] G. Rizzo, R. Troncy, S. Hellmann, and M. Bruemmer. NERD meets NIF: Lifting NLP extraction results to the linked data cloud. In *LDOW 2012, 5th Workshop on Linked Data on the Web, Volume 937, April 16, 2012, Lyon, France*, Lyon, FRANCE, 04 2012.

[17] H. Weizsäcker. *Catalog der Gemälde-Gallerie des Städelschen Kunstinstituts in Frankfurt am Main*. Frankfurt, August Osterrieth, Frankfurt am Main, 1903.