Chainable and Extendable Knowledge Integration Web Services

Felix Sasaki[1], Milan Dojchinovski[2,3] and Jan Nehring[1]

[1] Language Technology Lab
DFKI GmbH, Berlin, Germany
name.surname@dfki.de
[2] Knowledge Integration and Language Technologies (KILT/AKSW)
InfAI, Leipzig University, Germany
dojchinovski@informatik.uni-leipzig.de
[3] Web Intelligence Research Group
FIT, Czech Technical University in Prague
milan.dojchinovski@fit.cvut.cz

Abstract. This paper introduces the current state of the FREME framework. We will put FREME into the context of linguistic linked data and related approaches of multilingual and semantic processing. In addition, we focus on two specific aspects of FREME: the FREME NER e-Service, and chaining of e-Services. We believe that the flexible and distributed combination of e-Services bears a potential for their mutual improvement.

Keywords: Linguistic Linked Data, NIF, NLP, named entity recognition, workflows, multilingual and semantic enrichment

1 Introduction

This paper presents the current state of FREME, a framework for multilingual and semantic enrichment of digital content. A detailed, general overview of the goals of FREME has been given in [9]. Here we will focus on two aspects of FREME: the FREME NER service and chaining of FREME services.

FREME is developed in the EU funded FREME project[4], which started in February 2015 and lasts for two years. The project has two aspects: the development of the FREME framework, transferring technology outcomes from several language and data related projects; and the following four business cases:

1. Authoring and publishing multilingually and semantically enriched eBooks;
2. Integrating semantic enrichment into multilingual content in translation and localisation;
3. Enhancing the cross-language sharing and access to open agricultural and food data; and
4. FREME-empowered personalised content recommendations.

[4] See http://www.freme-project.eu

The paper is structured as follows. In section 2, we set FREME into the context of the KEKI workshop. In section 3, we provide a general overview of the FREME architecture. In section 4, we elaborate on the FREME NER service. In section 5, we discuss how this and other services can be chained together. In section 6, we conclude the paper.

2 FREME in Context

The development of the FREME framework can be described a) in the context of linguistic linked data, and b) with regards to challenges that arise from the four business cases.

Data related to linguistic and natural language processing. In the paradigm of linguistic linked data, more and more language resources are being published as part of the linguistic linked open data cloud[5]. FREME allows processing data available in the cloud as part of content enrichment workflows, for example to adapt named entity recognition with domain specific data sets.

Linguistic and NLP Ontologies. The LLOD cloud gatheres language resources that are represented with standard formats. FREME enrichment workflows make use of the following formats:

- The Natural Language Processing Interchange Format (NIF) [6] to represent data and enrichment information;
- The Internationalization Tag Set (ITS) 2.0[6] to represent metadata for improvement of enrichment workflows; and
- The OntoLex Lemon model [7] to represent lexica, including their meaning with respect to ontologies.

Linguistic linked open data workflows. The LLOD technology stack allows creating NLP and data services in a distributed and decentralized manner. FREME implements this stack by making use of the previously described standards, and by adding a declarative approach to define and re-use enrichment workflows.

NLP techniques for knowledge extraction. One aim of LLOD is to provide techniques for knowledge extraction that deploy linked data. FREME implements this approach in its FREME-NER service and allows users adapting the service with custom datasets, again to be provided as linked data.

Approaches using mappings and their maintenance from semistructured sources. Industry applications of NLP and data enrichment workflows have to deal with a plethora of content formats. Semistructured formats like HTML or certain XML formats are widely used in applications. Via its e-Internationalization service, FREME allows processing these formats, not only

[5] See `http://linguistic-lod.org/llod-cloud` for a latest version of the LLOD cloud.

[6] See `https://www.w3.org/TR/its20/`

[7] See `https://www.w3.org/2016/05/ontolex/`

for extraction, but for round-tripping, that is: storage of enrichment information in the original format.

The LLOD context of FREME can also be described from the point of view of the four business cases. From the business case perspective, several challenges arise when creating NLP and data processing applications. They are addressed in the following manner by FREME.

Interoperability and chainability. Applications often are provided as silo solutions. Integration of new functionality is then a time consuming task with high integration costs. By using the previously described, standardized technology stack, this effort is reduced significantly. Details are described in section 5.

Adaptability. There is a growing set of applications for key NLP tasks like named entity recognition, see e.g. [7]. Many of them rely on the DBpedia dataset [1] for entity linking. Tools like Stanford NER [5] allow users loading their own dataset and prepare it for NER. However, for users without a technological background in NLP, it is very hard to adapt these tools. FREME eases the adaptation process in several ways, with regards to the configuration of enrichment workflows, usage of custom data sets, and tailoring NER processing towards domains. Details for this adaptation are described in Section 4.

Data formats. The four business cases require enrichment workflows in many formats. For example, in localization, the XML based XLIFF format [8] is widely used. Current multilingual and semantic applications allow extraction of content from such formats. However, for real-life applications, the enrichment information has to be stored inside the format, without breaking existing processing tasks like validation, query or transformation. Via the e-Internationalization service, FREME allows such round-tripping processing.

2.1 Related Work

In this section we compare FREME to several related approaches: Apache Stanbol, Weblicht, Apache UIMA, and LAAPS Grid. They offer related capabilities and a comparison helps to understand the role of FREME.

Apache Stanbol offers a set of text analytics services in a Software as a Service manner. It is provided as an open source platform and intends to extend traditional content management systems with semantic services. In addition, the text analytics services can be used within arbitrary applications [2].

Apache Stanbol differs from FREME with regards to the set of services being offered. Although being open source and therefore being theoretically extensible, Apache Stanbol offers no detailed documentation on how to extend it. Further, Apache Stanbol puts a focus on the use case of a semantic content management system and semantic enrichment of homepages. Multilingual enrichment of other types of content is not taken into account. Further Apache Stanbol offers a variety of low level services like word splitting, part of speech tagging and more, so the

[8] See `http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html`

enrichment can be performed on different levels of granularity whereas FREME hides theses low level technologies to reduce the complexity of using the services.

Like FREME, *Weblicht*[9] offers support for chainable web NLP services in a RESTful manner. The main difference is that Weblicht does not constitute service chains via a linguistic linked data approach. This has the disadvantage that integration with linked data sources into Weblicht services needs an additional software integration step. In FREME no additional software integration is needed, since via the e-Link service, standard linked data query technology (SPARQL) can be deployed. Nevertheless, a conversion between the Weblicht native, XML based TCL format and the linguistic linked data format used in FREME should be possible and has the potential to grow the numbers of decentral NLP services.

APACHE UIMA [10] offers a framework for knowledge extraction pipelines. Like FREME, UIMA is extensible with various NLP components. A key difference to FREME again is that UIMA does not provide a linguistic linked data workflow for content enrichment. Instead, like Weblicht, UIMA provides an XML format. Another difference is that UIMA does not come with a Web service layer. This means that access to UIMA is specific to given programming languages (esp. Java or C++). In contrast, FREME can be accessed with nearly all programming languages, since the programming languages only have to offer HTTP request functionality. Since UIMA has a lot of existing modules, like in the case of Weblicht, a conversion between the UIMA and the NIF format could be of great value to benefit from existing NLP services. Further APACHE UIMA offers a variety of low level services like word splitting, part of speech tagging and more, so the enrichment can be performed on different levels of granularity whereas FREME hides theses low level technologies to reduce the complexity of using the services.

LAAPS GRID [11] is a framework that enables discovery, composition, and reuse of NLP components. LAPPS GRID comes with certain standards to support NLP tool interoperability. The LAPPS Interface Format (LIF) plays the role of NIF, that is, LIF constitues input and output of NLP workflows. The LAPPS Web Service Exchange Vocabulary defines the terms used in LIF, e.g. for parts of speech or other layers of linguistic annotation.

LIF is defined as a JSON format, which is a difference to the linguistic linked data approach taken by FREME. In addition, LIF and the terms defined by the vocabulary aim at fostering interoperability of the NLP detailed level processing, e.g. parts-of-speech, tokens etc. In FREME, this detailed level is not represented, but rather the higher level output of NLP processes, e.g.: annotated entities, translations, terms etc. This eases the integration with an application layer and integration with non-linguistic information, provided by the general linked data cloud.

[9] `http://weblicht.sfs.uni-tuebingen.de/`

[10] `http://incubator.apache.org/uima`

[11] `http://www.lappsgrid.org/`

The tools and initiatives discussed so far in this section all provide digital content processing functionality. A reviewer of the paper suggested also a comparison of FREME to META-SHARE [12]. META-SHARE is a distributed network of language resource repositories. In the future the FREME framework itself and resources generated via the FREME project will be made accessible via META-SHARE.

3 FREME Architecture

FREME uses a client-server Web service architecture that exposes Web services, called *e-Services*, via HTTP APIs. This approach allows for a decentralized, distributed creation of services in a RESTful architecture [4]. In this way a combination of services can be configured flexible insterad of being hard-wired in source code. Further the technology is not bound to a specific programming language, since almost every programing language supports HTTP based interactions [8]. Additionally, it is designed in an extensible manner, so that both project partners as well as external partners are able to add more services.

FREME uses common formats for language and data processing workflows, so that e-Services can easily be created by following the linguistic linked data technology stack. In this stack, the Natural Language Interchange Format (NIF) serves as a common broker format. Both the actual textual content and information generated via NLP and Linked Data processes is stored in NIF.

FREME offers six e-Services. Their functionality is summarized below.

- e-Entity offers named entity recognition. It is discussed in detail in section 4.
- e-Translation offers cloud based machine translation.
- e-Terminology offers enrichment of content with information about terms.
- e-Link offers enrichment with information from the linked data cloud.
- e-Publishing allows storing enriched content in the standardised ePub format.
- e-Internationalisation allows enrichment covering a wide range of digital content formats like HTML, generic XML or selected XML vocabularies.

Each e-Service is a pipeline on its own. For example, e-Entity consists of a series of tasks like word tokenization, part of speech tagging, sentence splitting and more. All these internal steps are hidden from the user. The user just submits text to the service and retrieves the entities. This lowers the complexity to use the service a lot. In some circumstances this might have a negative influence on the processing speed: When several e-Services are executed one after the other, some internal pipeline steps might be repeated.

In addition, the FREME framework is deployed in the German project ”Digitial Curation Technologies” (DKT) [13]. Services offered by DKT also use the linguistic linked data technology stack and hence can be combined with FREME services out of the box.

[12] `http://www.meta-share.eu/`

[13] See `http://digitale-kuratierung.de/` for details on the project

4 Content Enrichment with Names Entities

4.1 FREME NER Overview

E-Entity is one of the most exploited service within the FREME framework. Knowing what entities are mentioned in a document is of essential importance to better understand the aboutness of the document. The e-entity service annotates an input document with annotations representing entities. Mentions of entities, such as people, organizations or locations, are *spotted* and encoded with their position in the input document. Next, the entity is *disambiguated* by classifying from a set of entity types[14] and linking it to a specified knowledge base. The spotting and classification step is done by employing the StanfordNER tool [15][5] with a trained model on content from Wikipedia. The linking of entities ultimately relies on the most-frequent-sense approach and links with the most-frequent-sense entity. FREME NER is currently using models trained for English, German, Dutch, Spanish, Italian, French and Russian. To realize a MFS based linking we used Wikipedia as a reference knowledge base and collected every entity surface form, the corresponding hyperlink and the number of occurrences. As a result, a pair-count dataset [3] which provides this information was generated. The linking step is implemented in Apache Solr[16]. Solr contains indexed entities with their corresponding URI identifier, possible surface form variations, language, and the dataset they refer to. When performing the linking step, for an entity mention entity candidates are retrieved according to their surface form similarity, and the one with the highest `pair count` value is considered as the correct entity.

The listing 1.1 provides an example of the output from FREME NER.

```
<http://freme-project.eu/#char=0,33>
        a               nif:String , nif:Context , nif:RFC5147String ;
        nif:beginIndex  "0"^^xsd:int ;
        nif:endIndex    "33"^^xsd:int ;
        nif:isString    "Diego Maradona is from Argentina."^^xsd:string .

<http://freme-project.eu/#char=0,14>
        a                     nif:Word , nif:String , nif:Phrase , nif:RFC5147String ;
        nif:anchorOf          "Diego Maradona"^^xsd:string ;
        nif:beginIndex        "0"^^xsd:int ;
        nif:endIndex          "14"^^xsd:int ;
        nif:referenceContext  <http://freme-project.eu/#char=0,33> ;
        itsrdf:taClassRef     <http://dbpedia.org/ontology/SportsManager> , <http://
      dbpedia.org/ontology/Person> ... ;
        itsrdf:taConfidence   "0.9869992701528016"^^xsd:double ;
        itsrdf:taIdentRef     <http://dbpedia.org/resource/Diego_Maradona> .

<http://freme-project.eu/#char=23,32>
        a                     nif:String , nif:Word , nif:Phrase , nif:RFC5147String ;
        nif:anchorOf          "Argentina"^^xsd:string ;
        nif:beginIndex        "23"^^xsd:int ;
        nif:endIndex          "32"^^xsd:int ;
        nif:referenceContext  <http://freme-project.eu/#char=0,33> ;
```

[14] Currently, FREME classifies the entities with four types: PER, ORG, LOC and MISC for anything else.

[15] `http://nlp.stanford.edu/software/CRF-NER.shtml`

[16] `http://lucene.apache.org/solr/`

```
    itsrdf:taClassRef     <http://dbpedia.org/ontology/Place> , <http://dbpedia.org/
ontology/Location> ... ;
    itsrdf:taConfidence   "0.9804963628413852"^^xsd:double ;
    itsrdf:taIdentRef     <http://dbpedia.org/resource/Argentina> .
```

Listing 1.1. Output from FREME NER in the NIF format.

4.2 Entity Linking with Custom Datasets

In the last decade, entity linking has been primarily evaluated on datasets such as DBpedia, YAGO[17] and BabelNet[18]. In these use cases, the entity linking approaches have been exclusively customized to these datasets, and adoption of other datasets requires significant amount of effort, or it is not possible at all. In FREME, we allow users using their custom proprietary and public datasets and adopt the processing according to their needs.

FREME NER provides a dataset management endpoint which can be used to perform the usual dataset operations such as creation, update and deletion of a dataset. The minimum requirement is to provide a list of entities with a corresponding name variations. This information should be provided in RDF, where the subject of a triple is a URI, which uniquely identifies the entity, and the object is the entity name variation. The name variations can be provided using the RDFS[19] property `rdfs:label` or the SKOS[20] properties `skos:prefLabel` or `skos:altLabel`. While `rdfs:label` and `skos:prefLabel` specify the human-readable version of the entity name, `skos:altLabel` can provide alternative lexical labels for the entities. For example, a `pref:label` for the footballer Maradona is "Diego Armando Maradona", while `skos:altLabel` will be "Maradona".

Note that the confidentiality of proprietary datasets is ensured by implementing a secured access management. A user needs to be authenticated and authorized to get access to a dataset. Thus, only the owners of the particular datasets can consume them.

4.3 Domain Specific NER

In long texts, the list of recognized entities can be very large containing also entities which are not relevant to the domain of the document. For example, very often HTML content contains also advertisements from of text snippets which occur inline with the main content, or it contains entity mentions which are irrelevant for the main content. For example, an HTML document providing recent information about the Syrian crisis might encompass an advertisement related to the UEFA Euro 2016 championship, which mentions a football team or football player. FREME enables users to filter out such irrelevant entities by specifying the domain of interest (i.e. politics and administration) Thus,

[17] `http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/`

[18] `http://babelnet.org/`

[19] `https://www.w3.org/TR/rdf-schema/`

[20] `https://www.w3.org/TR/skos-reference/`

only entities from this specific domain will be returned. The implementation of this feature is realized by populating list of domains with corresponding entity types[21]. E.g. the types `dbo:PoliticalConcept` and `dbo:PublicService` belong to the domain of politics and administration.

Currently, FREME NER maintains "general" frequency counts information computed from the DBpedia Abstracts dataset [3]. These frequency counts information is used for the implementation of the most-frequent-sense based entity linking. In our future work, we plan to collect per-domain frequency counts and increase the performance of domain specific NER.

4.4 Experiments

In order to evaluate the 1) *quality* of the enrichments and the 2) *scalability* of the named entity recognition, we have conducted several experiments using GERBIL[22][11], a framework for evaluation of entity annotation tools. The experiments were executed using GERBIL version 1.2.3-SNAPSHOT via the live running instance at `http://gerbil.aksw.org/gerbil/`. The entity recognition was evaluated on five English collections and one German collection. The collections differ with regards to length of the documents, the density of entity mentions and the topic of the documents[23]. The evaluation was performed without performing domain specific NER. Two types of experiments were conducted in the evaluation. The strong annotation match requires exact match of the entity mention with the gold standard. The weak annotation match requires overlap of the entity mention with the annotation in the gold standard. Table 1 provides detailed results from the experiments for FREME NER. In Table 1 we report the macro and micro measures. The macro measures refer to the performance across the whole dataset, while the micro measures are computed for each documented and then averaged.

The results show that quality of the enrichments depends on the content. The best performance were achieved for the MSNBC dataset with `0.914` F1 for the weak annotation match and `0.805` F1 for strong annotation match. The worst performance has been achieved for the DBpedia Spotlight dataset with `0.349` F1 for the weak annotation match and `0.242` F1 for the strong annotation match.

In the experiments we have also evaluated the scalability of the entity recognition, and the evaluation results show that FREME NER in average can process one entity in 15 milliseconds or, in other words, 67 entities per second. Note that this conclusion should be taken with some reserve, since we implement caching. Hence, documents with frequently occurring entities will be processed faster. In [10] the authors report on the time needed to process a document for the MSNBC dataset. According to the results, except for the TagMe 2 system, for

[21] See the list of domains and related entity types at `https://github.com/freme-project/freme-ner/blob/master/src/main/resources/domains.csv`.

[22] `http://aksw.org/Projects/GERBIL.html`

[23] More information on the datasets is provided by [10].

Table 1. Detailed evaluation results of FREME NER.

Dataset	**Lang.**	**Exp. type**	**micro F1**	**micro P**	**micro R**	**macro F1**	**macro P**	**macro R**	**Millis per doc**	**Entities per doc**	**Millis per entity**
Spotlight	EN	weak	0.349	0.750	0.227	0.278	0.498	0.216	47.83	5.69	8.41
		strong	0.242	0.520	0.158	0.193	0.317	0.154	35.43	5.69	6.23
KORE50	EN	weak	0.956	0.940	0.972	0.957	0.958	0.975	31.98	2.86	11.18
		strong	0.894	0.879	0.910	0.890	0.888	0.909	30.52	2.86	10.67
Reuters-128	EN	weak	0.813	0.721	0.931	0.808	0.744	0.939	58.84	4.85	12.13
		strong	0.675	0.598	0.774	0.669	0.614	0.778	53.44	4.85	11.02
RSS-500	EN	weak	0.677	0.520	0.969	0.736	0.639	0.969	39.48	0.99	39.88
		strong	0.579	0.446	0.827	0.634	0.552	0.827	34.31	0.99	34.66
MSNBC	EN	weak	0.914	0.865	0.968	0.893	0.842	0.963	188.55	32.50	5.80
		strong	0.805	0.763	0.853	0.780	0.738	0.837	164.45	32.50	5.06
News-100	DE	weak	0.644	0.777	0.550	0.587	0.631	0.571	369.73	22.33	16.56
		strong	0.447	0.535	0.384	0.373	0.398	0.365	232.42	14.04	16.55

all the other systems it took more then one second to process an MSNBC document. In comparison, FREME NER required 914 and 805 milliseconds for the weak and strong annotation match, respectively.

In Table 2 we report on the performance of FREME NER compared to other six NER systems on the same set of datasets. We report only the micro F1 score for weak and strong annotation match type of experiment.

Table 2. Comparison of different NER systems and FREME NER.

Tool/Dataset	Exp. type	Spotlight	MSNBC	Reuters-128	KORE50	RSS-500
FREME NER	weak	0.349	**0.914**	0.813	**0.956**	0.677
	strong	0.242	**0.805**	0.675	**0.894**	0.579
DBpedia Spotlight	weak	0.413	0.559	0.512	n/a	0.422
	strong	0.392	0.481	0.331	0.493	0.359
Babelfy	weak	0.319	0.554	0.495	0.729	0.413
	strong	0.250	0.470	0.310	0.690	0.277
Entityclassifier.eu NER	weak	0.344	0.845	0.766	0.941	0.609
	strong	0.256	0.683	0.553	0.879	0.535
FOX	weak	0.222	0.348	**0.887**	0.833	**0.694**
	strong	0.189	0.029	**0.618**	0.784	**0.618**
NERD-ML	weak	**0.672**	0.632	0.484	0.760	0.391
	strong	**0.564**	0.534	0.374	0.728	0.267
TagMe 2	weak	0.663	0.454	n/a	0.766	0.521
	strong	n/a	n/a	0.305	n/a	0.354

The results show that for two datasets, MSNBC and KORE50, FREME NER achieved best performance. The results also show that for the RSS-500 and the Reuters-128 FREME NER achieved second best results, while for the DBpedia Spotlight dataset it achieved fourth best results. Note that we compared FREME NER to one of the most prominent NER systems such as DBpedia Spotlight, Babelfy, Enityclassifier.eu, FOX, NERD-ML and TagMe 2. Also note that we were not able to compute some scores for DBpedia Spotlight and TagMe 2 system, due to an unknown bug in those systems.

5 Chainable Web Services

As described previously, FREME NER is just one e-Service provided by the FREME framework. A key benefit of FREME is that its pipelining approach allows combination of e-Services. This will be explained with the example in listing 1.2.

```
{
"id": 55,
"description": "Example pipeline",
"serializedRequests": [
    {
    "endpoint": "http://api-dev.freme-project.eu/current/e-entity/freme-ner/documents",
    "parameters": {"language": "en"}
    },
    {
    "endpoint": "http://api.freme-project.eu/current/e-link/documents/",
    "parameters": {"templateid": "3"}
    },
    {
    "endpoint": "http://api-dev.freme-project.eu/current/e-terminology/tilde",
    "parameters": {
    "source-lang": "en", "target-lang": "nl" }
    },
    {
    "endpoint": "http://api-dev.freme-project.eu/current/e-translation/tilde",
    "parameters": {
    "source-lang": "en", "target-lang": "nl" }
    }
    ] }
```

Listing 1.2. Pipeline combining several e-Services

A pipeline consists of one or more steps. All steps are embedded in the serializedRequests JSON arrary. The order within the array defines the order of execution. Each step has a mandatory service endpoint and, depending on the endpoint, various optional or mandatory parameters. The steps can take various input formats. If, like in the example, no format is specified, a step assumes NIF input.

The first step in the example pipeline evokes FREME NER, which has been described in the previous section. The second step uses the e-Link service to retrieves information with a selected SPARQL query template. The template used in the example [24] retrieves geospatial information. The third step calls the e-Terminology service to enrich the content with terminology related information. This step needs a source and a target language, here English and Dutch. The last step calls the e-Translation service, with the same language pairs.

When services are chained, the chaining is controlled by the order of the steps. In the example, FREME NER is followed by e-Link, which is followed by e-Terminology and e-Translation. There is no separate workflow controller. The

[24] See `http://api.freme-project.eu/current/e-link/templates/3` to access the definition of the template.

data is sent between workflow steps without the need to explicitly interconnect them. This approach greatly simplifies the work for authors of pipelines.

Each step can take the default processing format for FREME as input: text/-turtle. It then processes the same format, without the need to explicitly declare the format for each step. For the first and the last step, that is, input and output of the pipeline, the pipeline author can declare formats explicitly. As of writing, HTML, XML, XLIFF 1.2 and ODT are are accepted as input.

The formats are not declared in the pipeline service itself, but as an HTTP content-type header in the service request. FREME then calls the e-Internationalisation service to process the format. In that way, a pipeline can be re-used with all formats. As output for roundtripping, currently HTML is accepted.

The example pipeline shows several benefits. First, one can compare the outcome of several e-Services. In the example, named entity recognition and terminology annotation are used to enrich the same content. This combination has the potential to improve both services via data based comparisons.

Second, there is no need to hardwire the combination of services, as long as the services adhere to the linguistic linked data stack. This can be seen in line 10 of the example. The e-Link service is installed on a different server (with the domain api.freme-project.eu) than the other e-services. The combination of services does not need a hardwired integration.

Third, the pipeline and in this way the e-Services are agnostic to given input and output formats. Format coverage is realised via the previously described e-Internationalisation service. Separating the actual services and the formats to be processed has the advantage that other services can easily be integrated and benefit from the growing set of formats being supported. The example in listing 1.3 shows how to call a pipeline with two alternative HTTP requests, executed via CURL.

```
curl -X POST -H "content-type: text/plain" -d 'Berlin is a nice city.'
"http://api.freme-project.eu/current//pipelining/chain/1"
curl -X POST -H "content-type: text/xml" -d 'Berlin is a nice city.'
"http://api.freme-project.eu/current//pipelining/chain/1"
```

Listing 1.3. Example CURL request.

The only difference between the requests is the content type header. In the second request, it is set to XML, which allows processing of general XML content.

If both input and output are set to the content type text/html, roundtripping becomes possible. That is, the enrichment information is stored in the actual HTML content. An example is given in listing 1.4. Here, a pipeline of first e-Entity, then e-Terminology has been applied to the HTML content. The result HTML contains dedicated attributes to store the term and entity related information.

```
<p>Welcome to the <span its-term-info-ref=
"http://example.com/#char=36,40"
its-term="yes">city</span> of <span
its-ta-class-ref=
"http://dbpedia.org/ontology/Settlement"
its-ta-ident-ref=
"http://dbpedia.org/resource/Prague">
```

```
Prague</span>.</p>
```

Listing 1.4. Pipeline with roundtripping of HTML.

We think that the combination of roundtripping and several e-Services has the potential to contribute to a data-driven comparision of e-Service outputs. In the example, there is information available from structured (HTML) markup, e-Terminology and e-Entity in the output. This representation makes a query trivial like: find all instances of entities which are in the same markup context as certain terms. The paragraph, represented as markup via the p element, would be a result for such a query, with the term city and the entity Prague. Such an interrelation of NLP output and original markup is not the aim of the current paper, but an interesting future topic of research.

Fourth, the pipelining greatly allows for automatization of repetivive processes and for making the content itself intelligent. For example, a client application could analyze the content with regards to the language of content and use this information for adapting the pipeline automatically.

6 Conclusion

This paper introduced the current state of the FREME framework with regards to two aspects: named entity recognition via the FREME NER e-Service, and chaining of e-Services. In addition, we have put FREME into the context of linguistic linked data and related approaches of multilingual and semantic processing.

The discussion on FREME NER showed some preliminary evaluation results. The pipeling of e-Services has a practical benefit (e.g. ease and automization of similiar language and data processing workflows), but also a research potential. We think that the combination of named entity recognition, terminology annotation and machine translation can lead to a data driven improvement of all of these technologies. This is a potential next step for FREME.

Acknowledgement. This work was supported via the FREME project funded from the EU Horizon 2020 research and innovation programme under grant agreement No 644771.

References

1. S. Auer, *et al.* DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web*, pp. 722–735. Springer, 2007.
2. R. Bachmann-Gmur. *Instant Apache Stanbol.* Packt Publishing Ltd, 2013.
3. M. Brümmer, M. Dojchinovski, and S. Hellmann. DBpedia Abstracts: A Large-Scale, Open, Multilingual NLP Training Corpus. In N. Calzolari, *et al.*, (eds.) *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016).* European Language Resources Association (ELRA), Paris, France, may 2016.
4. R. T. Fielding and R. N. Taylor. Principled Design of the modern Web Architecture. *ACM Transactions on Internet Technology (TOIT)*, 2(2):115–150, 2002.

5. J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local Information into Information extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 363–370. Association for Computational Linguistics, 2005.
6. S. Hellmann, J. Lehmann, S. Auer, and M. Brümmer. Integrating NLP using Linked Data. In *International Semantic Web Conference*, pp. 98–113. Springer, 2013.
7. P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. DBpedia spotlight: shedding Light on the Web of Documents. In *Proceedings of the 7th international Conference on Semantic Systems*, pp. 1–8. ACM, 2011.
8. C. Pautasso, O. Zimmermann, and F. Leymann. Restful Web Services vs. Big'Web Services: making the right architectural Decision. In *Proceedings of the 17th international conference on World Wide Web*, pp. 805–814. ACM, 2008.
9. F. Sasaki, *et al.* Introducing FREME: Deploying Linguistic Linked Data. In *Proceedings of the 4th Workshop on the Multilingual Semantic Web*. 2015.
10. R. Usbeck, M. Röder, and A.-C. N. Ngonga. Evaluating Entity Annotators using Gerbil. In *European Semantic Web Conference*, pp. 159–164. Springer, 2015.
11. R. Usbeck, *et al.* GERBIL – General Entity Annotation Benchmark Framework. In *24th WWW conference*. 2015.