

A Framework for the Needs of Different Types of Users in Multilingual Semantic Enrichment

Jan Nehring⁺, Felix Sasaki⁺⁺

⁺German Research Centre for Artificial Intelligence Berlin, Germany

⁺⁺Cornelsen, Berlin, Germany

jan.nehring@dfki.de, felix@sasakiatcf.com

Abstract

The FREME framework bridges Language Technologies (LT) and Linked Data (LD). It establishes workflows between LT and LD in a well defined, coherent way. FREME addresses common challenges that both researchers and industry face when integrating LT and LD: interoperability, "silo" solutions and the lack of adequate tooling. Usability, reusability and interoperability are often attributes of frameworks and toolkits for LT and LD. In this paper, we take a novel approach: We define user types and user levels and describe how they influence design decisions in a LT and LD processing framework. In this way, we combine research outcomes from various communities: language technology, linked data and software interface engineering. This paper explains the different user types and how FREME addresses the specific needs of each user type. Core attributes of FREME are usability, reusability and interoperability.

Keywords: language resource, language technology, usability, user types, linked data, software interface engineering

1. Introduction

Language Technology (LT) profits from the current rise of Artificial Intelligence and is being more and more integrated in commercial applications. The growing amount of digital content calls for technologies to process and enrich digital content in an automated manner. Both industry and academia need adequate tooling to process digital content. The amount of available tools is growing, but each tool uses its own data format. Merging annotations of digital content lacks a unified approach (Hellmann et al., 2013; Sanderson et al., 2013). Knowledge resources have become a key component of current systems in Artificial Intelligence (Flati et al., 2014). Construction and exploitation of such knowledge sources has gained attraction from both researchers (Mitchell, 2005; Navigli and Ponzetto, 2012) and big industry players like Google (Singhal, 2012) or IBM (Ferrucci, 2012). A popular approach to store and connect knowledge sources is Linked Data (LD).

The FREME framework bridges Language Technologies and Linked Data (Dojchinovski et al., 2016). It establishes workflows between LT and LD in a well defined, coherent way. FREME addresses common challenges that both researchers and industry face when integrating LT and LD: interoperability, "silo" solutions and the lack of adequate tooling (Sasaki et al., 2015a).

Usability, reusability and interoperability are often attributes of frameworks and toolkits for Language Technologies (LT) and Linked Data (LD). Examples are (Bachmann-Gmur, 2013; Hinrichs et al., 2010; Klein et al., 2017; Noji and Miyao, 2016). In this paper, we take a novel approach: We define user types and user levels and describe how they influence design decisions in a LT and LD processing framework. In this way, we combine research outcomes from various communities: language technology, linked data and software interface engineering. This paper explains the different user types and how FREME addresses the specific needs of each user type. Core attributes of FREME are usability, reusability and interoperability.

2. Background

This section first explains the different communities that are being addressed and bridged by the FREME framework. Then it explains the Natural Language Processing Interchange Format (NIF).

2.1. Bridging Language Resources, Language Technology and Engineering

Many approaches exist to integrate LD and LT, e.g. for Named Entity Linking (Ehrmann et al., 2016; Usbeck et al., 2014), Machine Translation (Srivastava et al., 2017) or Sentiment Analysis (Paul Buitelaar and Strapparava, 2013) but combining the two is still cumbersome and lacks a unified approach. Often LT profits from the combination with knowledge (Navigli and Ponzetto, 2012; Ristoski and Paulheim, 2016) but bridging language tools and knowledge sources in a well established, easy to use and coherent workflow is still a challenge. Researchers bridge LD and LT by storing the output of language tools using LD formats (Hellmann et al., 2013; Sanderson et al., 2013). Furthermore several challenges arise when these worlds meet, e.g. many different content formats to process; adaptability and avoiding "silo solutions"; usability and a lack of adequate tooling (Sasaki et al., 2015b). The FREME framework provides solutions to overcome these hardships.

2.2. The Natural Language Processing Interchange Format

FREME uses the Natural Language Interchange Format (NIF) as a common broker format to ensure that different LT and LD services are interoperable. NIF is an RDF/OWL based format that defines a common vocabulary to describe NLP annotations (Hellmann et al., 2013). NIF addresses the interoperability of NLP tools on three layers: It is based on Linked Data (structural layer) and a selection of ontologies to describe common NLP terms and concepts (conceptual layer). NIF aware applications are accessible via REST services (access layer). In the FREME terminology, such services are called e-Service.s

Every e-Service in FREME uses NIF or plaintext as input format and produces NIF as output format. Therefore, it is possible to feed the output of one e-Service in the next e-Service to form pipelines.

```
@prefix itsrdf: <http://www.w3.org/2005/11/its/rdf#> .
@prefix nif: <http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#> .
```

```
<http://freme-project.eu/#char=0,18>
  a nif:String , nif:RFC5147String ,
    nif:Context ;
  nif:beginIndex "0"
    ^^xsd:nonNegativeInteger ;
  nif:endIndex "18"
    ^^xsd:nonNegativeInteger ;
  nif:isString "Welcome to Berlin!"
    ^^xsd:string ;
  itsrdf:mtConfidence "0.725583686873588" ;
  itsrdf:target "Willkommen in Berlin!"@de .
```

Listing 1: Example of NIF for a translation

Listing 1 shows an example of NIF in the turtle serialization format. It shows the translation of an English document to German. The document is identified by a URL. This URL has further annotations, for example the textual content "Welcome to Berlin!" or the begin and end indices of the annotation. The translation (itsrdf:target) and the confidence value (itsrdf:mtConfidence) of the translation system are expressed via the ITRDF vocabulary.

3. User types

The design of FREME evolves around the following user types:

User level 0 is a technology specialist in digital content management, big data and multilingual and semantic technologies. He features specialised skills relevant to a certain domain. User level 0 can create FREME and its components. Innovation is generated at this user level and the outcomes are utilized by user levels 1, 2 and 3.

User level 1 is a data expert and a technology user. He has skills relevant to LD vocabularies and LD technologies. These users deploy FREME for working with concrete datasets. They build on the technologies developed by user level 0 and provide the basis for user levels 2 and 3. **User level 2** builds user interfaces and applications using FREME APIs and technologies provided by the lower user levels. His skill set is specialised on application development and knowledge of LT and LD technologies is usually low. The interface and application developer uses the high-level APIs provided by user level 1 to provide the basis for user level 3. Examples for user level 2 are web site architects or application developers.

User level 3 is the end user, e.g. a content creator, translator / localiser, publisher or others that create, process or consume content. He uses GUIs provided by user level 2 and is often not aware that the technologies he uses builds on FREME.

4. The FREME Framework

This section explains the architecture of FREME and its components in context of its user roles and the core at-

tributes usability, reusability and interoperability.

4.1. Architecture of FREME

The FREME framework exhibits a set of LT and LD services as HTTP APIs. FREME applications are client / server applications with the user being the client and the FREME API being the server. Figure 1 shows a birds-eye view on the architecture.

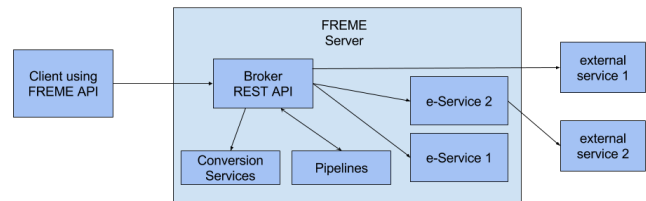


Figure 1: Architecture of FREME

The mandatory element of every FREME installation is the broker which acts as the entry point for every HTTP request. It redirects the request to the target e-Service or other module. An e-Service is a NLP enrichment service, e.g. Named Entity Recognition or Sentiment Analysis. Every e-Service uses NIF as input and output format. Each e-Service exhibits at least one HTTP endpoint that executes a certain LT or LD task. Every e-Service adds annotations to an existing NIF document. These annotations are called enrichments in the FREME terminology.

We chose this architecture for several reasons:

- The architecture allows a division of labour between language technology experts (user level 0-1) and application developers (user level 2). The language technology experts maintain the FREME server while the application developers merely use the service.
- Clients can use any programming language and any operating system.
- Further the clients can be lightweight because the server performs heavy processing.

(Sasaki et al., 2016) explain the architecture in more detail. The FREME framework offers the following e-Services out of the box:

- e-Entity for named entity recognition, classification and linking to the LD cloud
- e-Translation for statistical machine translation
- e-Terminology for terminology annotation

(Sasaki et al., 2016) and the FREME documentation give a more detailed overview about the services integrated in FREME. Other users of the framework provide additional e-Services. The project Digital Curation Technologies provides the following e-Services compatible with FREME (Rehm et al., 2017; Rehm and Sasaki, 2016):

- named entity recognition and linking

- event detection
- geographical localisation
- temporal expression analysis
- text classification
- text clustering
- monolingual and cross-lingual event detection
- single and multi-document summarisation
- machine translation

4.2. Concepts for different user levels

This section explains how the design of FREME helps the different user levels. This section explains benefits mainly for user level 1 and 2.

4.2.1. High level pipelines for easy integration

More and more LT tools exist but integrating these tools and managing inputs and outputs is still patchy. Each tool uses its own input and output format. Partially this problem is being addressed by NIF but FREME has an additional approach to ease the integration of LT and LD pipelines.

The individual steps of a traditional pipeline in the sense of e.g. UIMA (Ferrucci et al., 2009) or Gate (Cunningham et al., 2011) operate on a low level of abstraction, e.g. sentence splitting or tokenization. In FREME atomic steps operate on a high level of abstraction. So a single pipeline step can be e.g. Named Entity Enrichment, Machine Translation or similar. This means that a single step in a FREME pipeline often consists of a pipeline itself. These high level pipelines hide the low level complexity of the NLP tooling from the user. A single pipeline step is self-contained and does all the pre-processing of the data it needs itself. In this way, it is possible to use NLP services without detailed knowledge of their inner workings. Further one can rapidly develop FREME enabled pipelines and exchange processing steps at will. The NLP services do not pass the results of the low-level processing along the pipeline so this approach might lead to a lower performance since some steps may need to be done twice. So this approach is a trade-off between usability and performance.

The benefits for this approach are located at user level 1 and 2 because they can make use of the easy pipelines. User level 0 has to be aware of this concept so he can create services accordingly.

4.2.2. Accessing Linked Data Cloud without SPARQL

User level 2 usually has no expertise in using LD and SPARQL. Therefore, these technologies are often considered challenging. In order to be able to exploit the benefits of Linked Data without using SPARQL, FREME has introduced a division of labor between user level 1 and 2 using a mechanism called e-Link template. An e-Link template is a SPARQL CONSTRUCT query that is executed on a NIF document. It can for example extract all annotations of entities of type "city" in the NIF document and then enrich the document with museums or other tourist attractions located

in these cities. Another use case is to fetch additional information about persons like the birthday or a picture. These templates are stored by the FREME API. User level 1 defines these templates while user level 2 uses them. During the FREME project e-Link templates were used in several occasions and we found that they were very useful for both user levels because each could concentrate on what they do best and it established a clear workflow. Further it allows pipelining workflow without any hard-coded elements because defining e-Link templates requires configuration instead of programming. (Brümmer et al., 2016) explain the e-Link service in more detail.

4.2.3. Use Linked Data workflows without Linked Data knowledge using SPARQL filters

FREME has developed another concept to allow using LD based workflows without knowledge of LD. Although internally the pipelines use LD only, it allows other input and output formats.

FREME supports a wide range of input formats apart from LD, e.g. plaintext, HTML or PDF. In addition, it has a build in mechanism called SPARQL filter to convert LD output of the pipeline to a tabular format. A SPARQL filter is a SPARQL SELECT query that is applied to the NIF as the last step of the pipeline. A SPARQL SELECT query converts the graph based RDF data to tabular data which can be represented in Comma Separated Values, JSON or similar. User Level 1 defines the SPARQL converter, then user Level 2 uses it without getting in touch with Linked Data. Figure 2 shows an example of a pipeline that uses PDF as input format and tabular data as output format.



Figure 2: Example pipeline that uses PDF as input format and tabular data as output format

Another use case for the SPARQL filter API is simplification of the output. Pipeline output often contains a lot of information while only a subset of information is of interest. The SPARQL filter can reduce the size of the output and e.g. output person annotations only and strip all other information.

4.2.4. Use Linked Data workflows without Linked Data knowledge using e-Internationalization and XSLT

Using the XSLT API FREME can integrate in XML based workflows. The XSLT API can convert a document between several XML data formats using XSLT stylesheets that are stored on the server. E-Internationalization provides functionality to convert data from HTML5, which is also XML based, to NIF and back. Then NIF is converted to HTML5, enrichments will be embedded in HTML5 using the Internationalization Tag Set. Using this workflow, it is possible to define a pipeline that accepts XML as input and produces XML as output. This workflow is useful for user level 2. Figure3 shows an example of a pipeline that uses XML as input and output format.

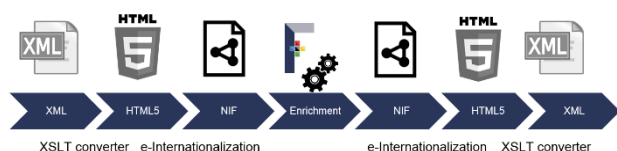


Figure 3: Example pipeline using XML as input and output format

4.2.5. Ready to use language resources, data resources and services

FREME offers a range of data services ready to use under an open license. This is useful for user level 1, although user level 2 can also benefit from the quick integration of these data services.

- Named entity recognition, classification and linking in 6 languages (English, German, Dutch, French, Italian, Russian) trained on the DBPedia Abstracts corpus (Brümmer et al., 2016)
- Datasets: DBPedia, Geopolitical Ontology, ORCID (Haak, Laurel L. and Fenner, Martin and Paglione, Laura and Pentz, Ed and Ratner, 2012), Statbel, Global Airports, Cordis, VIAF (Bennett et al., 2006), ONLD, GRID, FAO (Kim et al., 2013)
- e-Link templates: FREME offers a series of e-Link templates that can enrich NER annotations with information from the Linked Open Data cloud.
- SPARQL converters: We offer a series of converters to store RDF data as CSV for easier integration.
- FREME offers a set of common XSLT stylesheets to convert between XML formats using the XSLT converter service.

These LT and LD services are available through the official live instance of FREME. Further they can be downloaded and integrated in several ways into an on-premise FREME installation.

Several datasets were created or converted to Linked Data and integrated directly in the FREME framework. The rest of this section explains these datasets, in-depth information and downloads are located on Datahub¹. These datasets help user level 1 and 2 because they can be used without integration work.

The DBPedia Abstracts corpus (Brümmer et al., 2016) contains the abstracts of wikipedia articles in seven languages. The dataset provided training data for the e-Entity Named Entity Recognition service. Further it provides data for Named Entity Linking.

The Statbel corpus contains RDF conversion of datasets from "Statistics Belgium" which aims at collecting, processing and disseminating relevant, reliable and commented data on Belgian society.

OpenFlights.org contains a dataset about airport names, their locations, codes and other related info. The Global

Airports dataset is an RDF version of this data. In FREME it was used to enrich text for the tourism domain.

Open Researcher and Contributor ID (ORCID) is an index to uniquely identify scientific and other academic authors (Haak, Laurel L. and Fenner, Martin and Paglione, Laura and Pentz, Ed and Ratner, 2012). The 2014 version was converted to RDF.

The Global Research Identifier Database (GRID) contains information about worldwide research organizations. For FREME it was converted to RDF.

The Community Research and Development Information Service (CORDIS) is a dataset about EU funded research projects which was converted to RDF during the FREME project.

Other datasets that were already previously available and are now ready to use out of the box are VIAF (Bennett et al., 2006) and FAO (Kim et al., 2013).

5. Related Work

Apache Stanbol (Bachmann-Gmur, 2013), Weblicht (Hinrichs et al., 2010) and NLP curator (Clarke et al., 2012) are available as web services in a Software as a Service manner. The Unstructured Information Management Architecture (UIMA) (Ferrucci, 2012) has an extension to turn it into a web API (The Apache UIMA Development Community, 2008).

None of the above mentioned systems use a standardized data exchange format based on Linked Open Data. Since UIMA and Gate (Cunningham et al., 2011) are very popular their data formats have turned into quasi standards because of their wide adoption across several tools.

The Speech Analytics Platform (Batista et al., 2016) integrates several speech processing modules. It was developed with the aim to make usability of the modules as easy of possible. It has similar design principles as FREME: It is accessible as an API and provides a simple workflow to add new services. Further it can be used in a Graphical User Interface from the web browser.

Clarin Weblicht (Hinrichs et al., 2010) is similar to FREME because it also provides a web based execution environment and pipelines can span several APIs. Clarin Weblicht makes it easy to create pipelines in a web interface. It has a special emphasis on usability and interoperability because it targets users which do not have a technical background (Hinrichs and Krauwer, 2014). These users are from user levels 1 and 3.

The Jigg framework (Noji and Miyao, 2016) provides a methodology to integrate services for user level 1. Jigg suggests a workflow that relies on wrapping tools like Stanford Core NLP (Manning et al., 2014) with a Java objects, exchanging these objects in a pipeline and outputting the results in a proprietary XML format.

None of the above mentioned systems uses FREMEs concept of high level pipeline components. Pipelines in the sense of above systems are low level and therefore lack the easy integration.

Now the relation between FREME and other NLP tools is explored: There is a series of ready to use tools for certain NLP tasks, e.g. the widely used Stanford CoreNLP (Manning et al., 2014), Apache OpenNLP (The Apache Foun-

¹<https://old.datahub.io/dataset?tags=freme-project>

dation, 2017) or the Extended Named Entity Recognition API (Nguyen et al., 2017). The OpenMT toolkit (Klein et al., 2017) provides Machine Translation that has a similar focus on extensibility and modularity like FREME. These tools have a narrower focus than the FREME framework and can be integrated into FREME as e-Services. In this way only user level 0 is concerned with the inner workings and data formats of the tool. Subsequent user levels can utilize these tools like any other FREME e-Service and benefit from the FREME advantages.

Another background of FREME is research in interface engineering. Other research has identified three stakeholders affected by APIs and frameworks: API designers who implement the APIs. API users that implement applications using the work of the API designers and finally consumers that use the applications (Myers and Stylos, 2016). Research about API design argues that interoperability, reusability and interoperability are core features of APIs, along with other features such as learnability, security and expressiveness (Myers, 2017).

6. The FREME ecosystem

The FREME project was a two year innovation action funded by the Horizon 2020 program that ran from January 2015 - January 2017 under the lead of the German Research Centre for Artificial Intelligence. It consisted of a consortium of three research institutes and four industry partners. The goal was, among others, to help the partners bring multilingual, semantic technologies to the market. In the course of the FREME project the FREME framework described in this paper was developed. FREMEs business partners came from different domains of digital publishing, content recommendation, localization and internationalization and agriculture and food data.

6.1. The FREME community

The usefulness and the open nature of the framework attracted other users from outside of the consortium. The project Digital Curation Technologies (DKT) builds on the FREME framework also and adds new NLP services to the framework. Further the ADAPT research centre hosts a FREME server to provide a ready to use NLP API for its data scientists.

7. FREME in a real world scenario

The development of FREME was triggered by a number of use cases provided by the industry partners of the project. This section explains one of the use cases and the experiences made by this effort.

The use case was using FREME as part of the backend of a content recommendation platform. A backend service crawled websites and fed the documents into a FREME pipeline. Afterwards the features generated by the pipeline were used to recommend websites to the reader. The workflow used a clear separation of work between an application developer (user level 2) who developed the frontend and the data scientists (user level 1) who created the pipeline. The development of the backend from the data science side used rapid prototyping and different approaches were tried and compared to each other. During rapid prototyping the

high level services proved to be useful because it was easy and straight forward to change the NLP pipeline. Further the format coverage aspect of FREME proved to be useful because the HTML documents could be fed into the pipeline without time consuming preprocessing. At the end of the pipeline the SPARQL filter service converted the output from NIF to easy processable tabular data which contains only the important information from the NLP pipeline, stripping all unnecessary information.

During this project user level 0 created the underlying FREME services which are independent from the specific use case. User level 1 configured the services and provided the necessary knowledge sources. User level 2 could use the services without the need of a deep understanding of the underlying technology. User level 3 used the content recommendation without even noticing the FREME pipelines working in the backend. The user levels could not be totally separated. User level 0 and 1 often where the same people. Also user level 2 wanted to learn as much as possible about the underlying technology so he could be able to perform certain configuration tasks of the pipeline himself.

In this use case the differentiation between user levels and the impact of this differentiation on the design of FREME proved to be very useful because it established a clear workflow and every user level could focus on what he or she can do best. Especially the flexibility to quickly try out new approaches, mostly by configuration and without coding, was very useful.

8. Conclusion and Future Work

The aforementioned concepts proved to be useful to deploy applications that use both LD and LT in industry use cases. The FREME framework provides easy to use, reusable and interoperable services with a special focus on bridging the knowledge gaps between application developers and technology experts. The high level pipeline components made integration easy and flexible. Currently the work on the FREME framework focuses on the creation of new services and maintaining the core of the framework so it stays up to date. There are plans to augment FREME with big data processing capabilities. Other approaches focus on integrating FREME with cloud infrastructure providers like Amazon and executing FREME in a lambda function to create services that are scalable on demand.

Acknowledgments. This research was partially supported by the German Federal Ministry of Education and Research (BMBF) through the project BBDC (01IS14013E).

9. Bibliographical References

- Bachmann-Gmur, R. (2013). *Instant Apache Stanbol*. Packt Publishing.
- Batista, F., Curto, P., Trancoso, I., Abad, A., Ferreira, J., Ribeiro, E., Moniz, H., de Matos, D. M., and Ribeiro, R. (2016). {SPA:} Web-based Platform for easy Access to Speech Processing Modules. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation {LREC} 2016, Portoro{ž}, Slovenia, May 23-28, 2016*.
- Brümmer, M., Dojchinovski, M., and Hellmann, S. (2016). DBpedia Abstracts: A Large-Scale, Open, Multilingual

- NLP Training Corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).
- Clarke, J., Srikumar, V., Sammons, M., and Roth, D. (2012). An NLP Curator (or: How I Learned to Stop Worrying and Love NLP Pipelines). In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey.
- Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanić, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.
- Dojchinovski, M., Sasaki, F., Gornostaja, T., Hellmann, S., Mannens, E., Salliau, F., Osella, M., Ritchie, P., Stoitsis, G., Koidl, K., Ackermann, M., and Chakraborty, N. (2016). FREME: Multilingual Semantic Enrichment with Linked Data and Language Technologies. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4180–4183.
- Ehrmann, M., Nouvel, D., and Rosset, S. (2016). Named Entity Resources - Overview and Outlook. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).
- Ferrucci, D., Lally, A., Verspoor, K., and Nyberg, E. (2009). Unstructured Information Management Architecture ({UIMA}) Version 1.0. OASIS Standard.
- Ferrucci, D. A. (2012). Introduction to "This is Watson". *IBM J. Res. Dev.*, 56(3):235–249.
- Flati, T., Vannella, D., Pasini, T., and Navigli, R., (2014). *Two is bigger (and better) than one: The wikipedia bitaxonomy project*, volume 1, pages 945–955. Association for Computational Linguistics (ACL).
- Hellmann, S., Lehmann, J., Auer, S., and Bruemmer, M. (2013). Integrating NLP using Linked Data. In *The 12th International Semantic Web Conference (ISWC2013)*, pages 97–112.
- Hinrichs, E. and Krauwer, S. (2014). The {CLARIN} {Research} {Infrastructure}: {Resources} and {Tools} for e-{Humanities} {Scholars}. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 1525–1531.
- Hinrichs, E., Hinrichs, M., and Zastrow, T. (2010). WebLicht: Web-based LRT Services for German. In *Proceedings of the ACL 2010 System Demonstrations, ACLDemos '10*, pages 25–29, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *CoRR*, abs/1701.0.
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The {Stanford} {CoreNLP} Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.
- Mitchell, T. (2005). Reading the Web: A Breakthrough Goal for AI. [\url{http://www.cs.cmu.edu/~tom/pubs/AImagazine-7-2005.html}](http://www.cs.cmu.edu/~tom/pubs/AImagazine-7-2005.html).
- Myers, B. A. and Stylos, J. (2016). Improving API Usability. *Commun. ACM*, 59(6):62–69.
- Myers, B. A. (2017). Human-centered Methods for Improving API Usability. In *Proceedings of the 1st International Workshop on API Usage and Evolution, WAPI '17*, page 2, Piscataway, NJ, USA. IEEE Press.
- Navigli, R. and Ponzetto, S. P. (2012). {B}abel{N}et: {T}he Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Artificial Intelligence*, 193:217–250.
- Nguyen, T. D., Mai, K., Pham, T.-H., Nguyen, M. T., Nguyen, T.-V. T., Eguchi, T., Sasano, R., and Sekine, S. (2017). Extended Named Entity Recognition {API} and Its Applications in Language Education. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, {ACL} 2017, Vancouver, Canada, July 30 - August 4, System Demonstrations*, pages 37–42.
- Noji, H. and Miyao, Y. (2016). Jigg: {A} Framework for an Easy Natural Language Processing Pipeline. In Sameer Pradhan et al., editors, *Proceedings of {ACL-2016} System Demonstrations, Berlin, Germany, August 7-12, 2016*, pages 103–108. Association for Computational Linguistics.
- Paul Buitelaar, Mihael Arcan, C. A. I. J. F. S.-R. and Straparava, C. (2013). Linguistic linked data for sentiment analysis. In *2nd Workshop on Linked Data in Linguistics (LDL- 2013): Representing and linking lexicons, terminologies and other language data*, Pisa, Italy.
- Rehm, G. and Sasaki, F. (2016). Digital Curation Technologies. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation (EAMT'2016)*, Riga, Latvia.
- Rehm, G., Moreno Schneider, J., peter Bourgonje, Srivastava, A., Nehring, J., Berger, A., König, L., Räuchle, S., and Gerth, J. (2017). Event Detection and Semantic Storytelling: Generating a Travelogue from a large Collection of Personal Letters. In *Proceedings of the Events and Stories in the News Workshop*, pages 42–51, Vancouver, Canada, aug. Association for Computational Linguistics.
- Ristoski, P. and Paulheim, H. (2016). Semantic Web in data mining and knowledge discovery: A comprehensive survey. *Web Semantics: Science, Services and Agents on the World Wide Web*, 36:1–22.
- Sanderson, R., Ciccacese, P., and de Sompel, H. V. (2013). Designing the W3C open annotation data model. In Hugh C Davis, et al., editors, *WebSci*, pages 366–375. ACM.
- Sasaki, F., Gornostay, T., Dojchinovski, M., Osella, M., Mannens, E., Stoitsis, G., Ritchie, P., and Koidl, K. (2015a). Introducing FREME: Deploying Linguistic Linked Data. In *The 4th Workshop on the Multilingual Semantic Web: co-located with ESWC 2015, Portorož, Slovenia*.

- Sasaki, F., Gornostay, T., Dojchinovski, M., Osella, M., Mannens, E., Stoitsis, G., Ritchie, P., and Koidl, K. (2015b). Introduction to FREME: Data meets Language meets Business. In *The 1st International Workshop on the Use of Multilingual Language Resources in Knowledge Representation Systems (MLKREP2015)*, Vienna, Austria.
- Sasaki, F., Dojchinovski, M., and Nehring, J. (2016). Chainable and Extendable Knowledge Integration Web Services. In *First Workshop on Knowledge Extraction and Knowledge Integration - ISWC 2016*.
- Singhal, A. (2012). Introducing the knowledge graph: things, not strings. *Official Google Blog*, May.
- Srivastava, A., Rehm, G., and Sasaki, F. (2017). Improving Machine Translation through Linked Data. *The Prague Bulletin of Mathematical Linguistics*, 108(1):355–366, jan.
- The Apache Foundation. (2017). Apache OpenNLP.
- The Apache UIMA Development Community. (2008). UIMA Simple Server User Guide.
- Usbeck, R., Ngomo, A.-C. N., Röder, M., Gerber, D., Coelho, S., Auer, S., and Both, A. (2014). AGDISTIS - Graph-Based Disambiguation of Named Entities Using Linked Data. In Peter Mika, et al., editors, *The Semantic Web – ISWC 2014*, volume 8796 of *Lecture Notes in Computer Science*, pages 457–471. Springer International Publishing.

10. Language Resource References

- Bennett, R., Hengel-Dittrich, C., O’Neill, E. T., and Tillett, B. B. (2006). VIAF (Virtual International Authority File): Linking Die Deutsche Bibliothek and Library of Congress Name Authority Files.
- Brümmer, M., Dojchinovski, M., and Hellmann, S. (2016). DBpedia Abstracts: A Large-Scale, Open, Multilingual NLP Training Corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).
- Haak, Laurel L. and Fenner, Martin and Paglione, Laura and Pentz, Ed and Ratner, H. (2012). ORCID: a system to uniquely identify researchers. *Learned Publishing*, 25(4).
- Kim, S., Iglesias-Sucasas, M., and Viollier, V. (2013). The FAO Geopolitical Ontology: A Reference for Country-Based Information. *Journal of Agricultural & Food Information*, 14(1):50–65.