MULTISENSOR

Mining and Understanding of multilinguaL contenT for Intelligent Sentiment Enriched coNtext and Social Oriented inteRpretation

FP7-610411

D5.4

Final semantic infrastructure and final decision support system

Dissemination level:	Public
Contractual date of delivery:	Month 34, 31 August 2016
Actual date of delivery:	Month 36, 4 October 2016
Work package:	WP5 Semantic Reasoning and Decision Support
Tasks:	T5.1 Knowledge modelling T5.2 Semantic representation infrastructure
Type:	Prototype
Approval Status:	Final Draft
Version:	2.0
Number of pages:	17
Filename:	D5.4_FinalSemanticInfrastructureDecisionSupport_2016-10-04_v2.0.pdf

Abstract

This document presents the MULTISENSOR RDF application profile, the methodologies and pipeline deployment of Dependency Parsing for Bulgarian, as well as the semantic infrastructure integration, i.e. how easily the system can be adapted to different languages and domains.

The information in this document reflects only the author's views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

Co-funded by the European Union

History

Version	Date	Reason	Revised by
0.1	01/08/2016	Document initiation	B. Simeonov (Ontotext)
0.2	22/08/2016	First iteration with contributions	B. Simeonov, V. Alexiev, K. Simov, V. Kotsev (Ontotext)
0.3	05/09/2016	Second iteration with contributions	B. Simeonov, V. Alexiev, K. Simov, V. Kotsev (Ontotext)
0.4	24/09/2016	Integrated document	B. Simeonov, V. Alexiev (Ontotext)
0.5	28/09/2016	Internal review	G. Meditskos (CERTH)
1.0, 2.0	03/10/2016	Final version	B. Simeonov (Ontotext)

Author list

Organisation	Name	Contact Information
Ontotext	Boyan Simeonov	boyan.simeonov@ontotext.com
Ontotext	Vladimir Alexiev	vladimir.alexiev@ontotext.com
Ontotext	Kiril Simov	kiril.simov@ontotext.com
Ontotext	Venelin Kotsev	venelin.kotsev@ontotext.com

Executive Summary

This document presents the MULTISENSOR RDF application profile, the methodologies and pipeline deployment of Dependency Parsing for Bulgarian, as well as the semantic infrastructure integration, i.e. how easily the system can be adapted to different languages and domains.

Abbreviations and Acronyms

API	Application Programing Interface
CSV	Comma Separated Value
DSS	Decision Support System
FTS	Full-text Search
GDP	Gross Domestic Product
IP	Internet Protocol
ITSRDF	Internationalisation Tag Set Ontology
JSON	JavaScript Object Notation
JSONLD	JavaScript Object For Linked Data
NER	Named Entity Recognition
NIF	NLP Interchange Format
NLP	Natural Language Processing
OLAP	Online Analytical Processing
OLiA	Ontologies For Linguistic Annotation
POS	Part Of Speech
RDF	Resource Description Framework
SIMMO	Socially Interconnected and MultiMedia-enriched Object
SME	Small And Medium Entrepreneur
SPARQL	SPARQL protocol and RDF query language
UC	Use Case
UI	User Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WN	Wordnet

Table of Contents

1 INTRODUCTION

The document is organised as follows: Section 2 presents the MULTISENSOR RDF application profile. The profile includes information about the used ontologies, standards, the developed SPARQL queries and evaluation according to the self-assessment plan in deliverable D1.2. Section 3 describes the methodologies and pipeline deployment of Dependency Parsing for Bulgarian (reported in this deliverable as additional work relevant also to WP2). Section 4 describes the semantic infrastructure integration, i.e. how easily the system can be adapted to different languages and domains. Finally, some concluding remarks are provided in Section 5.

2 MULTISENSOR RDF APPLICATION PROFILE

The MULTISENSOR project analyses and extracts data from mass- and social media documents, including text, images and video, across several languages. It also handles social network data, statistical data, etc.

Early in the project, a decision was made to capture all data exchanged among the project partners in RDF JSONLD (Sporny, 2014) format. The final data is stored in a semantic repository and is used by various User Interface components for end-user interaction. This final data forms a corpus of semantic data over SIMMOs (news and social network objects), which constitutes an important outcome of the project.

The semantic data flexibility allowed us to accommodate a huge variety of data in the same extensible model. We used a number of ontologies for representing this data: NIF and OLIA for linguistic info, ITSRDF for NER, DBpedia and Babelnet for entities and concepts, MARL for sentiment, OA for image and cross-article annotations, W3C CUBE (Cyganiak, 2014) for statistical indicators, etc. In addition to reusing existing ontologies, we extended them with the MULTISENSOR ontology, which introduced some innovative concepts, such as embedding FrameNet conceptualisations in NIF.

The documentation of this data is an important task, as the MULTISENSOR data is meant to be efficiently used by external consumers.

The "living document"[1] describes the different RDF patterns used by MULTISENSOR and how the data fits together. Thus, it represents an "RDF Application Profile" for MULTISENSOR. We use an example-based approach, instead of the more formal and laborious approach being standardised by the W3C RDF Shapes working group (still work in progress). We have found that examples are easier to understand both for data providers (project partners that produce the various pieces of data) and for data consumers (developers that need to write queries).

We cover the following areas:

- Linguistic Linked Data in NLP Interchange Format (NIF), including Part of Speech (POS), dependency parsing, sentiment, Named Entity Recognition (NER), etc.
- Speech recognition and translation.
- Multimedia binding and image annotation.
- Statistical indicators and similar data.
- Social network popularity and influence, etc.

2.1 MULTISENSOR SPARQL queries

The companion document MULTISENSOR SPARQL Queries[2] describes various queries used by the system. We found this an indispensable collaboration tool while developing the various queries required by the system. It can also be used as an example tutorial by the external consumers of the data. It includes 66 queries, organised as follows:

[1] Formerly https://github.com/VladimirAlexiev/VladimirAlexiev.github.io/tree/master/Multisensor, now at https://github.com/VladimirAlexiev/Multisensor

[2] https://docs.google.com/document/d/1FfkiiTYvrLzHJ5P5j34NRVGPbXml0ndpNtiNbH2osRw/edit

Section	**Queries**
Getting Info from Single SIMMO	10
Getting Info From Many SIMMOs	1
Multimodal Search	4
Semantic Search	3
Statistical Queries	38
EuroStat Exploration	4
EuroStat Queries	6
Format Date According to Freq.	3
Economic indicators	10
Social indicators	5
Political indicators	3
Cultural indicators	3
Data Quality	4
Named Entity Data	4
Abstractive Summarisation	6
Total	**104**

Table 1: MULTISENSOR SPARQL queries

2.2 Ontologies self-assessment

We conducted an informal self-assessment regarding the adequacy of the ontologies and application profile to the tasks of the MULTISENSOR project (see deliverable D1.2, page 30):

- The relevance and expressiveness of ontologies adopted or developed specifically for the project is very high, with some minor adjustments. Some examples:
    - The NLP Interchange format (NIF), with NIF 2.1 extensions, was able to capture many aspects of MULTISENSOR NLP data, including provenance, confidence, and annotations by several tools.
    - To embed FrameNet in NIF, some innovation was required.
    - It was easy to apply Open Annnotation (OA) to the Content Alignment service (which expresses whether two SIMMOS express similar or contradictory views).

- The quality and richness of ontologies describing external datasets varies with the dataset. Some examples:
    - We have found that the Babelnet ontology is able to describe Babelnet entities accurately (e.g. labels, broader concepts, Wikipedia category links, etc.). We explored the latter two for reasoning about related concepts in order to provide an enriched search.
    - Wikipedia categories exhibit strong "semantic drift", i.e. the further you navigate from an original concept along broader categories, the less is the chance that the concepts found are relevant to the original context. That is why we use only 1-2 levels of categories.
    - The DBpedia ontology and data have various data quality issues (e.g. the parents of people are not always people, due to the way links are extracted from infoboxes). We are engaged in the DBpedia Quality Committee and are aware of them.

3 BULGARIAN DEPENDENCY PARSING

In this Section, we present the implementation of the MULTISENSOR Bulgarian pipeline, which involves some additional work from WP2. This service is implemented as a separate pipeline on a different server than the ones used for the rest of WP2 services. For this reason, these achievements are reported in the current deliverable. The pipeline includes the following steps: tokenization, sentence splitting, POS and grammatical features tagging, lemmatisation and dependency parsing.

3.1 Approach overview

In order to implement the Bulgarian pipeline, the hybrid approach has been followed. Some of the components are rule-based and some are statistical-based.

The rule-based approach is used for tokenization, sentence splitting and lemmatisation. The statistical-based approach is used for POS and grammatical tagging, as well as for dependency parsing. The statistical models are trained over data from Bulgarian Treebank - BulTreeBank. The morphosyntactic annotation includes a rule-based and a statistical module. The rule-based module corrects some of the suggestions from the statistical one.

The conversion of the linguistic analysis to ontology representation is done by mapping the morphosyntactic and dependency tagsets of BulTreeBank to OLIA, PenTreebank and MultextEast ontologies.

3.2 Implementation in MULTISENSOR

The POS tagging in Bulgarian is more complex than in English. Bulgarian is also an analytical language, but with rich word inflection. Although we often refer to this task as POS tagging, it is more accurate to be defined as morphosyntactic annotation or morphological tagging, because of the big variety of grammatical features and their interdependence. To tackle the complexity of the problem in an adequate way, we use the full form of 680 tags of the BulTreeBank Morphosyntactic Tagset (BTB-TS)[3], which is the original tagset of the BulTreeBank. Its positional encoding of different morphosyntactic features allows us to better train statistical models for tagging and parsing, as it provides the most important linguistic features of the word forms.

Rule-based Module for Morphological Tagging

The rule-base module for morphological tagging exploits two sources of linguistic knowledge: the morphological lexicon (nearly 110000 lemmas) and the gazetteers (more than 26000 names), as well as the set of 70 disambiguation rules, implemented in CLaRK System. The rules are handcrafted and then arranged as an algorithm, as described below. The rules are extensively tested during the creation of BulTreeBank and they produce 100% accuracy result.

The rules work on an input, in which the tokens are annotated with all possible tags provided by the morphological lexicon. First, the algorithm looks up the morphological dictionary and retrieves all possible tags for each token in the text. Then, the rules can

[3] http://www.bultreebank.org/TechRep/BTB-TR03.pdf

narrow down the possible tags for a given word by selecting one of the possible tags. In the rest of the cases, all possible tags remain in the annotation. They were designed to achieve higher precision even at the cost of low recall.

MATE tools for POS tagging and dependency parsing

For statistical POS tagging and parsing, we are using the MATE tools[4]. The MATE tools are considered the best state-of-the-art tools for these tasks. The lexicons and the tools from the previous step are used as a filter for the result from MATE POS tagger. Whenever the lexicons and rules predict a tag that is not suggested by the statistical tagger, we use it. The accuracy for POS is 97.72%. The UAS for dependency parser is 92.9%. Both modules were specially trained on a new version of the Bulgarian treebank. Within the project, we are annotating domain data with morphological information and dependency analyses. A part of this data will be used for retraining of these modules for the final version. Another part will be used for domain evaluation of the modules.

Lemmatisation

The lemmatisation module comprises a set of transformation rules that we developed, based on the morphological lexicon. They were implemented via finite state automata in the CLaRK system instead of word forms directly being looked up in the lexicon. We motivate our decision with its faster operation speed. Furthermore, the rules were based on the morphological dictionary, presented above. The accuracy of the lemmatiser is a little above 95%.

Ontology Mapping

The part-of-speeches, grammatical features and dependency relations are mapped to the available ontologies for Linguistic Linked Open Data.

BulTreeBank Morphosyntactic Tagset (BTB-TS)[5] is a position-based tagset for Bulgarian. It is similar to MulText East tagset (MTE-TS), but it differs in granularity. For example, some features, which are encoded in MulText East tag, are not accepted as relevant for the Bulgarian language and thus they are not encoded in BTB-TS, while others are represented in BTB-TS but not in MTE-TS. Therefore, we do not directly reuse the mapping of MTE-TS to its ontological representation, but instead we have created our own mapping. Example of the correspondences can be seen in Table 2:

Ncmsi	mte:CommonNoun, mte:Noun, mte:MasculineGender, mte:SingularNumber, mte:Indefinite
Ansd	mte:Adjective, mte:NeuterGender, mte:SingularNumber, mte:Definite
Vpiif-r3p	mte:MainVerb, mte:Verb, mte:Intransitive, mte:Indicative, mte:PresentTense, mte:ThirdPerson, mte:PluralNumber

Table 2: MTE-TS mapping

[4] https://code.google.com/p/mate-tools/

[5] http://www.bultreebank.org/TechRep/BTB-TR03.pdf

Although Bulgarian is a morphologically rich language and the tagset is large (680 tags), we also provide a mapping to the PenTreebank tagset (PT-TS), which could facilitate the usage of some tools that are tuned to PT-TS. Much of the information here is lost. For example, all nine forms of the Bulgarian adjectives are mapped to a single tag: penn:JJ.

Finally, in the dependency version of BulTreeBank, we have 18 dependency relations. All of them are mapped to corresponding class in OLiA ontology. For example, BTB dependency relation 'indobj' is mapped to olia:IndirectObject.

The following Figures depict a dependency tree, as it is produced by the Bulgarian pipeline, as well as its ontological representation.

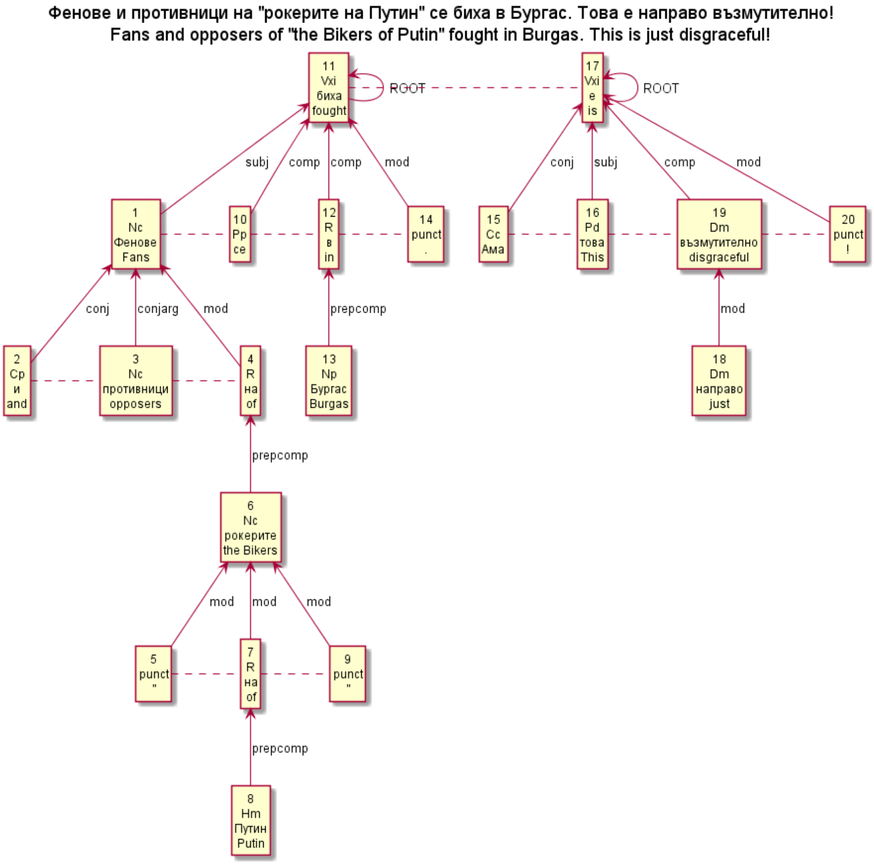

Figure 1: Dependency tree produced by the Bulgarian pipeline

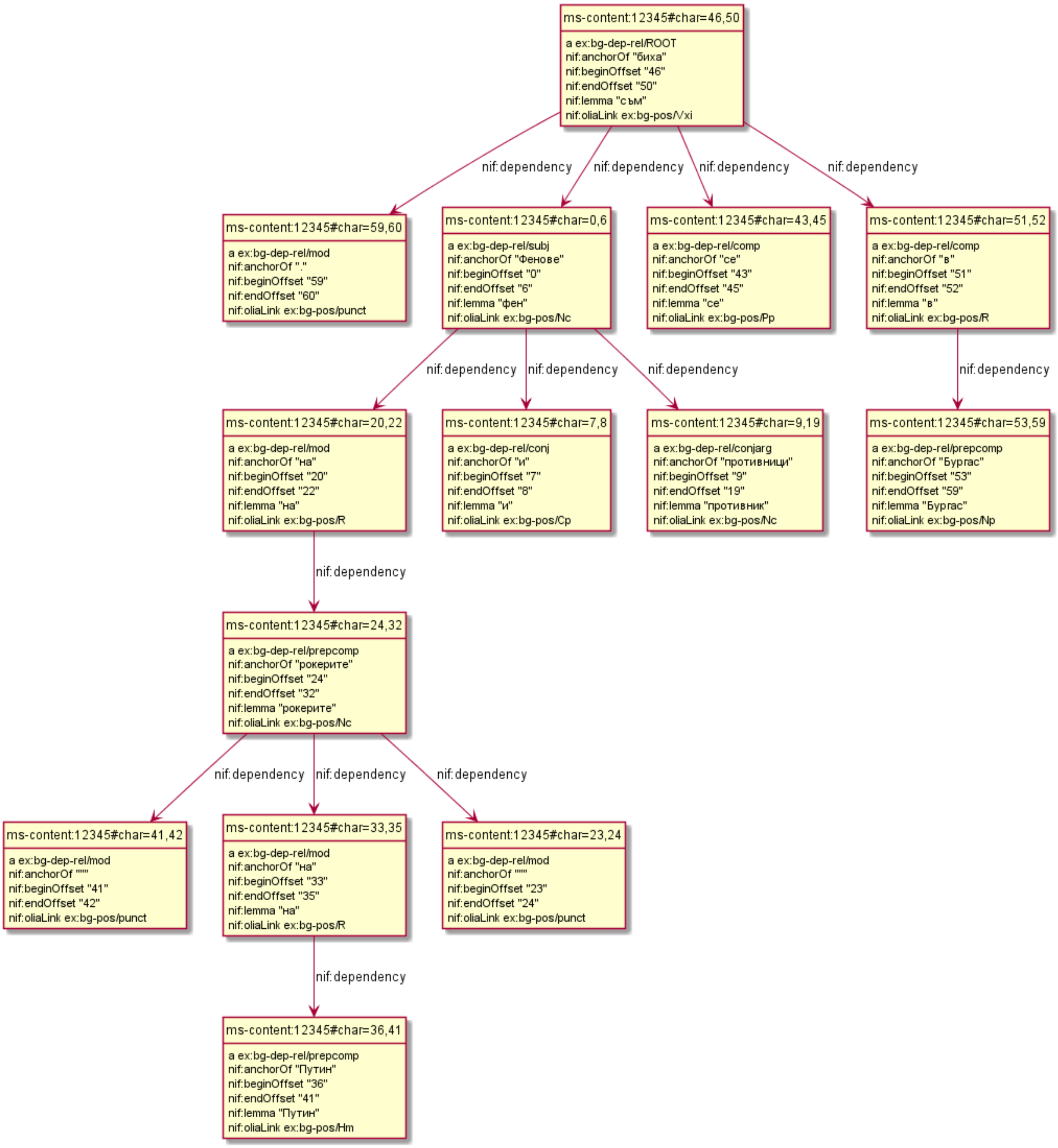

Figure 2: Ontological representation of dependency tree (produced by the Bulgarian pipeline)

4 SEMANTIC INFRASTRUCTURE INTEGRATION

System portability and adaptation is a very important aspect in the lifecycle of all software systems. In projects like MULTISENSOR, the ability of the system to work with multilingual content is crucial. Due to the fact that GraphDB is triple store, it is fully language independent. The multi-language support, in this case, is provided by the services that produce the stored data and the ones that consume it. Part of the semantic infrastructure are the following services and APIs:

- RDF Storing service;
- Decision Support System framework;
- Hybrid Search API.

The RDF Storing service works as a layer over GraphDB. Its purpose is to handle SIMMO objects, parse them to the appropriate format - RDF, and store them in the knowledge base. Such a layer is necessary, because the structure of the SIMMO is divided into two parts - a simple JSON object and a JSONLD encoded array. The storing service has to merge these two parts together and produce a valid RDF. According to the fact that its main objective is to parse JSONs, we can claim that it is fully language independent. The only restriction is that the received input should be valid.

The Decision Support System developed for MULTISENSOR is built on top of GraphDB. In its core are statistical indicators. All indicators are modelled with the CUBE ontology. At the current state of the system, we can easily add new indicators to the knowledge base but the framework will need some modifications in order to handle new data. The algorithms behind the system are adapted to UC3, which means that they take into account values like distance, product type and country of origin. In other words, some new development will be needed in order to adapt the system to a new domain. The required effort will vary in the different cases and domains. Apart from that, the system is fully language independent.

The Hybrid search API is designed to work with language independent data. All filters and queries are build runtime, according to the input parameters. This make the API fully language independent. However, there are other specifics. The API is built on top of GraphDB. The service is hardly dependent on Elasticsearch connected through GraphDB connectors. This leads to a conclusion that in order to integrate the API into another system, we will need GraphDB engine as a base.

5 CONCLUSIONS

In this deliverable, we have presented the MULTISENSOR RDF application profile, in which an example-based approach is used, instead of the more formal and laborious approach being standardised by the W3C RDF Shapes working group. In addition, we have presented the implementation of the MULTISENSOR Bulgarian pipeline, which involves some additional work from WP2. The pipeline includes the following steps: tokenization, sentence splitting, POS and grammatical features tagging, lemmatisation and dependency parsing. Finally, we have described the semantic infrastructure integration, i.e. how easily the system can be adapted to different languages and domains.

6 REFERENCES

Beckett, D., Berners-Lee, T., Prud'hommeaux, E. and Carothers, D. RDF 1.1 Turtle: Terse RDF Triple Language. W3C Recommendation 25 February 2014. **https://www.w3.org/TR/turtle/.**

Capadisli, S., Auer, S. and Riedl, R. Towards Linked Statistical Data Analysis. Proceedings of the 1st International Workshop on Semantic Statistics, Sydney, Australia, October 11th, 2013. CEUR Volume 1549, urn:nbn:de:0074-1549-5. http://csarven.ca/linked-statistical-data-analysis.

Cyganiak R., DERI,Galway N., Reynolds D., Epimorphics Ltd 2014. “The RDF Data Cube Vocabulary”. **https://www.w3.org/TR/vocab-data-cube/**.

Harris S., Garlik, Seaborne A., The Apache Software Foundation 2013. “SPARQL 1.1 Query Language”. https://www.w3.org/TR/sparql11-query/.

Sporny M., Bazaar D., Longley D., Bazaar D., Kellogg G., Associates K., Lanthaler M., Graz University of Technology, Lindström N. 2014. “JSON-LD 1.0”. https://www.w3.org/TR/json-ld/.

ter Horst, H. J.: Combining RDF and Part of OWL with Rules: Semantics, Decidability, Complexity. In Proc. of ISWC 2005, Galway, Ireland, LNCS 3729, pp. 668-684, November 6-10, (2005)