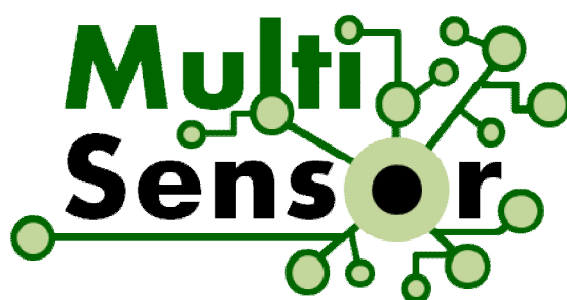




MULTISENSOR

Mining and Understanding of multilingual content for Intelligent Sentiment
Enriched context and Social Oriented interpretation

FP7-610411

D2.2

Basic techniques for speech recognition, text analysis and concept detection

Dissemination level:	Public
Contractual date of delivery:	Month 12, 31/10/2014
Actual date of delivery:	Month 12, 31/10/2014
Workpackage:	WP2 Multilingual and multimedia content extraction
Task:	T2.2 Named entity extraction workflows T2.3 Concept extraction from text T2.4 Concept linking and relations T2.5 Audio transcription and analysis T2.6 Multimedia concept and event detection T2.7 Machine translation
Type:	Report
Approval Status:	Final Draft
Version:	1.0
Number of pages:	78
Filename:	D2.2_BasicSpeechRecognitionConceptDetection_2014-10-31_v1.0.pdf

Abstract

This deliverable reports on the basic techniques for text analysis, audio speech recognition, machine translation, entity recognition and multimedia concept detection developed in the tasks T2.2, T2.3, T2.5, T2.6 and T2.7.

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	11/07/2014	Draft	V. Aleksić (LT)
0.2	30/07/2014	Document structure refined	V. Aleksić (LT)
0.3	30/09/2014	Contributions from UPF to sections 4 and 5	G. Casamayor (UPF)
0.4	30/09/2014	Contribution from Linguattec to sections 3, 6, and 8	G. Thurmair, V. Aleksić (LT)
0.5	02/10/2014	Contributions from CERTH to section 7	A. Moumtzidou, S. Vrochidis (CERTH)
0.6	02/10/2014	First consolidated draft	V. Aleksić (LT)
0.7	16/10/2014	Feedback from WP2 partners to the consolidated draft.	V. Aleksić (LT), A. Moumtzidou (CERTH), G. Casamayor (UPF)
0.8	21/10/2014	Corrections according to the feedback (extended conclusions, added missing acronyms, reviewed and implemented suggestions)	V. Aleksić (LT), A. Moumtzidou (CERTH), G. Casamayor (UPF)
0.9	21/10/2014	Corrections consolidated; document ready for the internal review	V. Aleksić (LT)
0.10	27/10/2014	Internal review	K. Simov (ONTO)
1.0	28/10/2014	Final version after feedback from internal review by Kiril Simov	V. Aleksić (LT)

Author list

Organisation	Name	Contact Information
Linguattec	Vera Aleksić	v.aleksic@linguatec.de
Linguattec	Gregor Thurmair	g.thurmair@linguatec.de
UPF	Gerard Casamayor	gerard.casamayor@upf.edu
CERTH	Anastasia Moumtzidou	moumtzid@iti.gr
CERTH	Stefanos Vrochidis	stefanos@iti.gr

Executive Summary

This deliverable reports on the basic techniques for speech recognition, machine translation, and multimedia concept detection, as well as the text analysis pipeline which in turn includes the recognition and extraction of named entities and concepts from text, as well as links and relations between them.

The document describes in detail all modules in WP2, their approaches, components, and resources. For each module an overview of the state of the art and a comparison to other approaches is included. Furthermore, the corresponding infrastructure and the integration of the modules into the MULTISENSOR system are described. Where applicable, the process of the creation of training and testing datasets is described, the evaluation approaches are explained and the evaluation results presented. The deliverable presents the work done during the first year of the project.

Specifically the following modules are presented: a) the named entities recognition module, which identifies names of persons, locations, companies and institutions, as well as amounts and dates; b) the concept extraction and concept linking and relation extraction modules from textual information; c) the speech recognition module, which is based on open-source framework RWTH-ASR; d) the concept extraction module from images and video; e) the machine translation module.

Abbreviations and Acronyms

AP	Average Precision
API	Application programming interface
ARPA	Advanced Research Projects Agency
AS	Automatic Summarisation
ASR	Automatic speech recognition
BLEU	Bilingual Evaluation Understudy
BoW	Bag of Words
CMLLR	Constrained Maximum Likelihood Linear Regression
CSV	Comma-separated values
DGT	Directorate-General for Translation (EU)
DSyntS	Deep syntactic dependencies structures
DUL	Dolce + DnS Ultralite
FV	Fisher Vector
G2P	Grapheme to Phoneme
GATE	General Architecture for Text Engineering
GB	Gyga Byte
HMM	Hidden Markov Model
HSV	Hue, Saturation, and Value
HTK	Hidden Markov Model Toolkit
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IE	Information extraction
IR	Information retrieval
IRSTLM	Information Retrieval Source and Target Language Modelling
JPEG	Joint Photographic Experts Group (Image file format)
JRC	Joint Research Centre (EU)
JSON	JavaScript Object Notation
JSON-LD	JSON for Linked Data
JSONP	JSON (JavaScript Object Notation) with padding
LDA	Linear Discriminant Analysis
LM	Language Modelling

LOD	Linked Open Data
LLC	Limited Liability Company
LT	Linguatec
LTD	Limited, as in ACME, Ltd.
LVCSR	Large Vocabulary Continuous Speech Recognition
MB	Mega Byte
MFCC	Mel Frequency Cepstral Coefficients
MPEG	Moving Picture Experts Group (Audio/visual file format)
NE(R)	Named Entities (Recognition)
NED	Named Entities Disambiguation
NIF	NLP Interchange Format
NIST	National Institute of Standards and Technology
NLG	Natural Language Generation
NLP	Natural Language Processing
OCR	Optical character recognition
OOV	Out of Vocabulary
OWL	Web Ontology Language
PCA	Principal Component Analysis
PDF	Portable Document Format
PLP	Perceptual Linear Prediction
PossNE	Possible Named Entity (Named Entity candidate)
RBMT	Rule-based Machine Translation
RDF	Resource Description Format
REST	Representational State Transfer
RGB	Red, Green, and Blue
RSS	Really Simple Syndication (or Rich Site Summary)
RWTH	Rheinisch Westfälische Technische Hochschule (German: Rhenish-Westphalian Technical University; Aachen, Germany)
SIFT	Scale-invariant feature transform
SIN	Semantic Indexing Task
SME	Small and medium-sized enterprises
SMT	Statistical Machine Translation
SRI	Speech Recognition Interface

SRILM	SRI Language Modelling
SRL	Semantic Role Labeller
SSynts	Surface syntactic dependency parser
SURF	Speeded Up Robust Features
SVM	Support vector machine
TAZ	Tageszeitung (German newspaper)
TRECVID	TREC Video Retrieval Evaluation
UC1	Use Case 1
UC2	Use Case 2
URI	Uniform Resource Identifier
UIMA	Unstructured Information Management Applications
URL	Uniform Resource Locator
UVI	Unified Verb Index
VLAD	Vector of Locally Aggregated Descriptors
VTLN	Vocal Tract Length Normalisation
WER	Word Error Rate
WP	Work Package
WSD	Word Sense Disambiguation
XML	Extensible Markup Language

Table of Contents

1	INTRODUCTION	11
2	ARCHITECTURE OF THE CONTENT EXTRACTION MODULE (WP2)	12
2.1	Content extraction pipeline.....	12
2.2	Formats and services.....	13
3	NAMED ENTITIES RECOGNITION	14
3.1	Management summary	14
3.2	State of the art	14
3.3	Approach	15
3.3.1	Linguistic Objects	15
3.3.2	Architecture	15
3.4	Implementation	15
3.4.1	Interfaces.....	15
3.4.2	Software	16
3.5	Resources	16
3.5.1	Lexicon.....	16
3.5.2	Grammars.....	17
3.5.3	Coreference	18
3.6	Integration into the MULTISENSOR platform and status.....	18
4	CONCEPT EXTRACTION FROM TEXT	19
4.1	Management Summary.....	19
4.2	Approach.....	19
4.3	Software.....	20
4.4	Resources	21
4.5	Evaluation	22
4.6	Integration into the MULTISENSOR platform.....	22
5	CONCEPT LINKING AND RELATIONS	23
5.1	Management Summary.....	23
5.2	Approach.....	23
5.3	Software.....	24
5.4	Training resources.....	24
5.5	Integration into the MULTISENSOR platform	25
6	AUDIO TRANSCRIPTION	26
6.1	Management summary	26

6.2	State of the art	26
6.3	Technical framework and approach	26
6.4	Implementation and components	27
6.4.1	Data preparation	27
6.4.2	Language models	29
6.4.3	Pronunciation dictionary	29
6.5	Evaluation	30
6.6	Integration into the MULTISENSOR platform	30
7	TECHNIQUES FOR MULTIMEDIA CONCEPT DETECTION	31
7.1	State of the Art in Multimedia concept detection	31
7.1.1	Video decoding	31
7.1.2	Feature extraction	32
7.1.3	Classification	33
7.2	Approach	34
7.2.1	Video decoding	34
7.2.2	Feature extraction	35
7.2.3	Classification	35
7.3	Modules description	35
7.3.1	Video decoding	36
7.3.2	Feature extraction	36
7.3.3	Classification	37
7.4	Concept selection for MULTISENSOR use cases	38
7.4.1	Journalism use case scenario	38
7.4.2	Commercial media monitoring use case scenario	40
7.5	Creation of training dataset	40
7.6	Evaluation	41
7.6.1	Time performance evaluation	42
7.6.2	Model evaluation	44
8	MACHINE TRANSLATION	48
8.1	Management summary	48
8.2	State of the art	48
8.3	Technical framework and approach	49
8.4	Implementation and components	49
8.4.1	Resources	50
8.4.2	Data preparation	50
8.5	Evaluation	52
8.6	Integration into the MULTISENSOR platform	52
9	CONCLUSIONS	53
10	REFERENCES	55

A	APPENDIX: ANNOTATION GUIDELINES	65
A.1	Introduction	65
A.2	Annotation of nominal expressions	66
A.3	Annotation of verbal expressions	68
B	APPENDIX: NAMED ENTITIES RECOGNITION - SPECIFICATION	71
B.1	Entity types	71
B.2	Objects and attributes.....	72
B.2.1	Document	72
B.2.2	Entities.....	72
B.2.3	Entity type structures	72
B.3	Output example as JSON object	75
C	APPENDIX: TECHNIQUES FOR MULTIMEDIA CONCEPT DETECTION.....	77
C.1	Creation of training dataset	77
C.2	Evaluation	77

1 INTRODUCTION

This deliverable reports on the work done in WP2 of the MULTISENSOR project during the first project year. The objective of WP2 is to extract knowledge from multimedia input data, and present it in a way that later components can operate on them.

The current report comprises all tasks of WP2, except of T2.1 that was successfully completed in month 6 of the project and described in D2.1 (Empirical study on media monitoring and internationalisation resources). Accordingly, it describes the work done in tasks T2.2 (Named entity extraction workflows), T2.3 (Concept extraction from text), T2.4 (Concept linking and relations), T2.5 (Audio transcription and analysis), T2.6 (Multimedia concept and event detection), and T2.7 (Machine translation).

As such, these tasks contribute to the milestone MS2 of the project (the completion of the setup of the operational infrastructure of the MULTISENSOR system) and correspond to the first year (Y1) activities A2.1 to A2.6, described in the project roadmap D7.1 as shown in Figure 1.

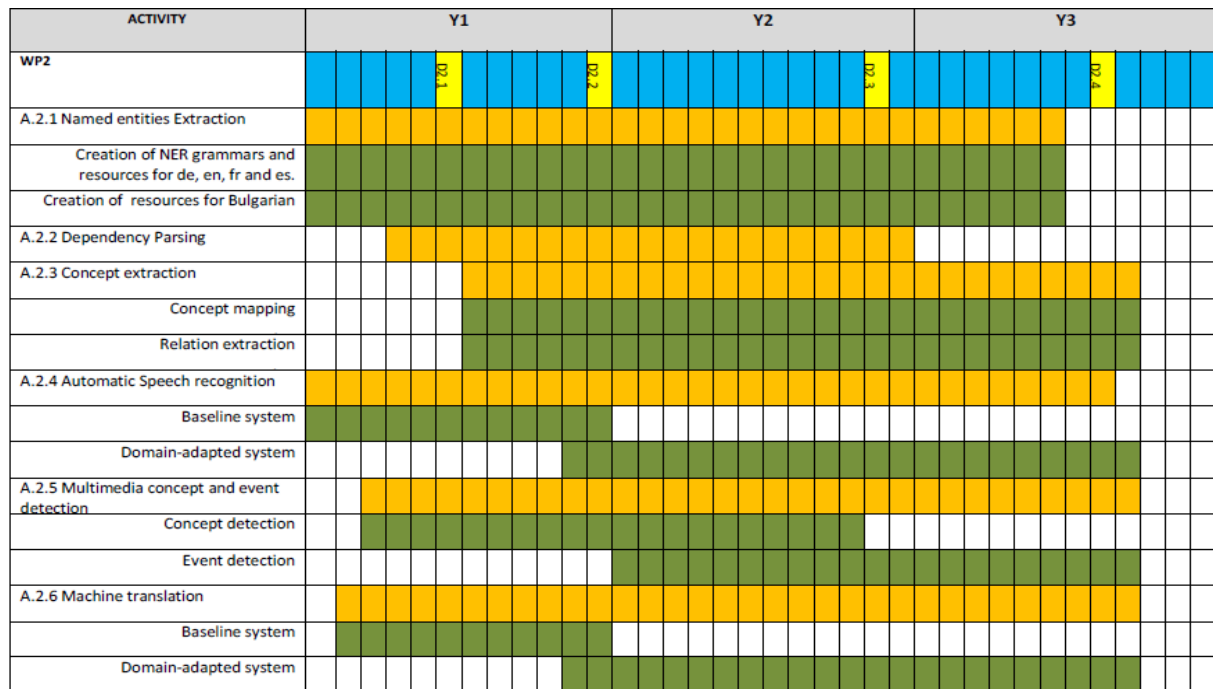


Figure 1: WP2 activities and timeline.

Each task is described in a different section of this document (Sections 3, 4, 5, 6, 7, and 8). The introductory Section 2 gives an overview of the analysis pipeline and the general architecture of WP2. In Section 9 we draw some conclusions about the work done in WP2 during the first twelve months of the project.

2 ARCHITECTURE OF THE CONTENT EXTRACTION MODULE (WP2)

2.1 Content extraction pipeline

The objective of WP2 "Multilingual and Multimedia Content Extraction" is to extract knowledge from multimedia input data (audio, text, video, image). These raw input data are turning into "knowledge", when the key information is recognised, processed and organised in a way that allows the user to retrieve it, reason on it, draw conclusions and make decisions.

In WP2, this is done by consecutively running a series of content analysis and extraction services:

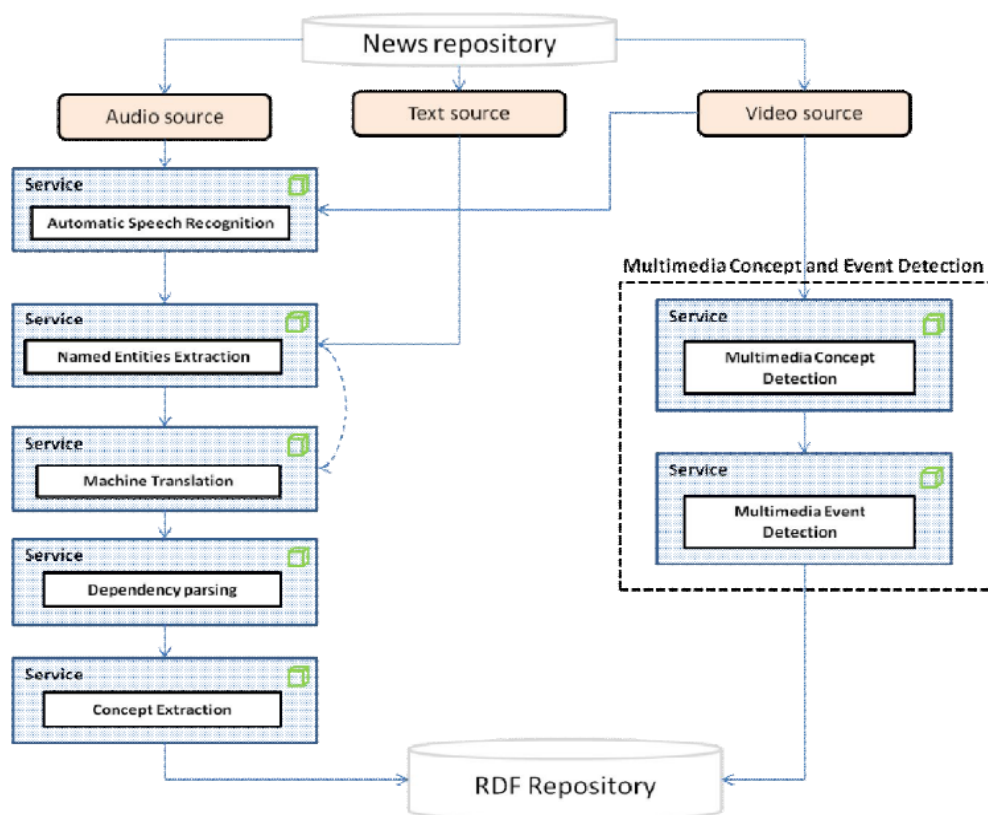


Figure 2: WP2 pipeline.

The recognised and extracted knowledge (named entities, other textual concepts, relations between them, as well as multimedia concepts and events) are presented in a structured way (as RDF statements) and stored in the RDF repository. The knowledge extraction pipeline is ontology-oriented and an essential part of the knowledge base creation is the ontology "population". A set of target ontologies are used to model the contents extracted from the text. This consists in identifying the key information in text and multimedia and relating them to concepts in the ontology¹. In contrast to the ontology population, the

¹ cf. also D5.1 „Basic semantic infrastructure“

ontology learning approach consists in identifying new concepts and relations not modelled by the target ontologies.

All subsequent components (content integration and retrieval, semantic reasoning, and abstractive summarisation) can then operate on the information extracted and organised in this way.

2.2 Formats and services

All WP2 modules will be serviced and hosted independently of each other, either on the common MULTISENSOR platform hosted by Everis, or at proprietary platforms of other partners, who are providers of the services, depending on the performance and computing power requirements of the respective module.

The services are deployed as REST² web services. They communicate with each other via public APIs. Messages are exchanged using the JSON³ format. Other formats are used to encode certain types of output within the main JSON message. Services that produce textual annotations use the RDF/OWL-based NIF⁴ ontology. In addition to NIF, linguistic annotations are encoded using the OLiA⁵ ontologies (also RDF/OWL-based). All RDF output is further encoded using the JSON-LD⁶ format and stored in a specific part of the main JSON message.

More information can be found in D7.1 “Roadmap towards the implementation of MULTISENSOR platform” and D7.2 “Technical requirements and architecture design”.

² cf. http://en.wikipedia.org/wiki/Representational_state_transfer

³ <http://www.json.org/>

⁴ <http://persistence.uni-leipzig.org/nlp2rdf/>

⁵ <http://www.acoli.informatik.uni-frankfurt.de/resources/olia/>

⁶ <http://json-ld.org/>

3 NAMED ENTITIES RECOGNITION

3.1 Management summary

The objective of the Named Entities recognition is to identify names (named entities) in texts. **Names** are words which uniquely identify objects, like ‘Maastricht Treaty’, ‘Berlin’, ‘Siemens’. Of course names can be ambiguous, i.e. denote several things, like ‘Barcelona’ pointing to a city or a region.

Names belong to different **types**: Companies, locations, time expressions, and many others. In its first version, the Linguatrec NE recognition component identifies the following entity types:

- persons
- locations
- organisations, divided into companies and institutions
- amounts
- dates

There will be sub-categories for some of the major categories (e.g. type of location, prices as subclass of amounts, etc.). The languages covered for the NER task in the MULTISENSOR project will be English (en), German (de), Spanish (es), Bulgarian (bg), and French (fr).

3.2 State of the art

Named Entities recognition (NER) is a task which has been researched for more than a decade now. It has been extended along several dimensions.

1. In the dimension of named entities **recognition**, new domains like medical (Tepper et al., 2012), bioinformatics (Bundschuh et al., 2008), or stock price predictions (Lee et al., 2014) have broadened the scope of NE analysis beyond the news domain, and applications like sentiment analysis (Declerck & Krieger, 2014) extend NE techniques.

Progress also went from NE recognition to NE linking (also called NE Disambiguation (NED)), where the entities are linked to known entities stored in a NE database (McNamee et al., 2010; Sameshiba Taba & Caseli, 2014); and to slot filling, i.e. finding annotations to entities in texts, and creating full annotated records of entities in an effort to populate knowledge bases. Challenges for slot filling lie in linguistic analysis (cf. Min & Grishman, 2012). Research of this kind is carried out in the context e.g. of the Text Analytics Conference (TAC 2009ff)⁷.

Temporal analysis and structuring has soon been found to be of special importance for NE analysis. Activities like TempEval (Verhagen et al., 2009) focused on this aspect. Again, it turned out that a change in domain (beyond news) has significant effects on the technology to be used for temporal analysis (Strötgen & Gertz, 2012).

2. Another dimension of NE research is **multilinguality**. Previous attempts (Steinberger, 2010; Thurmair, 2004) were also extended in the context of named entity linking (Li et al., 2012; Lawrie et al., 2012), but also into aligning news texts in different languages based on

⁷ <http://www.nist.gov/tac>

the NEs which they contain (Barker & Gaizauskas, 2012). In MULTISENSOR, dealing with multilingual input, the link between Named entities recognition and machine translation is of particular importance; multilingual entity linking based on string similarity and transliteration will not be sufficient, especially in case of locations.

3. A third dimension is the setup of a **processing chain**, or a workflow for a sequence of text extraction modules. Approaches range from GATE (one of the earliest frameworks, Cunningham, 2000; Cunningham et al., 2010) and UIMA-based systems (Banky & Schierle, 2012) to flows of web services as e.g. explored in PANACEA (Bel et al., 2012).

3.3 Approach

The difference to “normal” lexicon entries is that names are an open class, and cannot simply be recognised by lexicon lookup. Therefore, the recognition consists of two parts: lexicon lookup for entities in a lexicon, and additional identification means for entities not in the lexicon. The challenge is to identify, and correctly classify (type assignment) these entities.

3.3.1 Linguistic Objects

Many NE components, like those based on the GATE and UIMA frameworks, build on shallow analysis techniques (tokeniser, tagger). Such approaches have limited reach and are difficult to extend in cases where additional linguistic tasks are required.

Therefore, the approach in MULTISENSOR is to choose a technology which can be extended. The approach is to allow for deep analysis of texts, where NER is based on the manipulation of complex linguistic objects instead of strings; these objects are described by feature structures. NE recognition creates and manipulates such feature structures.

3.3.2 Architecture

The NE component consists of three software components: Sentence splitting, tokenisation, and NE recognition. The NE recognition uses three components: local parser, text analyser (for coreference determination etc.), and output generation.

While the software components are basically language-independent, there are resource files which provide the data for the respective languages. Two main resources are used: an NE lexicon (an annotated gazetteer), and a grammar (set of grammar rules).

3.4 Implementation

3.4.1 Interfaces

The NE component is accessed via an HTTP protocol. The request is analysed by a server component (written in the Ruby programming language⁸) which calls the respective Java⁹ executable for analysis, and returns the analysis result.

⁸ <https://www.ruby-lang.org/en/>

⁹ <https://www.oracle.com/java/index.html>

For data input, MULTISENSOR partners have agreed on input in pure text format and using the UTF-8 encoding.

The result of the recognition will be provided in two flavours:

- An XML markup format (like ,xxx <person>Peter Muller</person> xxx').
- A JSON format where all found entities are collected in a JSON structure. This format is used in the MULTISENSOR processing chain.

The result is returned using the HTTP protocol mentioned above.

3.4.2 Software

The main software components of the system are the following:

- A **sentence splitting** component, segmenting a text into its sentences.
- A **tokenisation** component. The tokeniser delivers a list of tokens, each annotated with (normalised) text form, onset, offset, and casing.
- The **NE recognition**. It identifies NE objects in the input token list. Such objects can be multiwords (consisting of multiple tokens).

The NE recognition consists of three major components: the local parser, the text analyser, and the output generator.

- The **parser** is implemented as an active chart like in (Winograd, 1986; Charniak, 1997).
- The **text analyser** has the task to link NE candidates into the same entity.
- The **output generator** collects all occurrences of an entity, and provides them in one of the output flavours just mentioned.

The recogniser is implemented in Java.

3.5 Resources

The system uses two types of resources (several smaller lists, like abbreviation files etc., are neglected): an NE lexicon, and an NE recognition grammar.

3.5.1 Lexicon

Structure

The system uses special lexicons. The lexical material is organised along two lines:

- **Language-related** data comprises the lexicons for the respective languages. Language-specific entries can be found not just for location names (Moscow (en), Moscou (fr), Moskau (de)) but also for institutions (NATO (en), OTAN (fr)), companies and even person names (e.g. in transliterations: Gorbatchev (en), Gorbatschow (de)).
- **Task-related** data is stored in different sublexicons. Each language has three sublexicons:
 - A **name** lexicon, containing the lexicalised named entities, like the most frequent/important person, company, place names, units, date names (like ,Christmas') etc.
 - An **indicator** lexicon, collecting words which are used as indicators for entity types, like 'city_of', 'Prof.', 'car maker', 'lives_in' and so on. Most of these indicators are language-specific.

- A domain lexicon which is used to tune the recogniser for a particular domain. This lexicon will be created from domain-specific resources.

The lexicon parts are compiled into a single file, which is then used at runtime.

Content

The data for the name lexicon need to follow certain principles. There is no point in dumping all names of Wikipedia into the lexicon; such an approach has been tried and abandoned. Therefore the following principles have been applied:

- For **persons**, the most frequent last names per language have been collected from the internet (between 500 and 1000 names). In the language-specific lexicons, names in the respective language were added.
- For **locations**, all countries, and the biggest cities worldwide and in Europe were selected. In addition, smaller regional cities, counties / region names and other geographic names were added for German, English, Spanish etc.
- The **company names** listed in the stock exchanges of the European, US and Asian stock markets were collected. They were not separated by language.
- For **institutions**, a (multilingual) glossary provided by JRC¹⁰ has been examined and parts of the entries have been reused.

The resulting lexicon contains about 50K names per language. This is a basic set which is kept rather small deliberately:

It should be noted that using frequency as selection criterion makes the resource strongly dependent on the domain-specific corpus used. For a generic tool, this is not really a good strategy, and makes the NE components strongly domain-dependent. However, for domain adaptation, text analytics will have to be used, and the domain specific names will be added to the lexicon to improve accuracy.

3.5.2 Grammars

The name recogniser uses grammars for NE detection and type assignment.

Rule format

The grammar rules are phrase-structure rules, augmented by scoring information and by feature operations. They consist of an identifier, a phrase structure part, a probability score, a feature handling section, and an example and comment field.

Feature handling is done by operators, like unifications, but also other feature tests, percolation instructions, feature setting etc.

Analysis Strategy

On a top level, the grammar distinguishes between NEs and non-NEs (Txt nodes); sentences are analysed recursively from the left, by consuming NE and Txt nodes as they come along.

On a lower level, the grammar first provides a special rule set for lexicalised NEs and their multiword parts. In addition, it identifies possible NEs (PossNE) from patterns of unknown.

¹⁰ <https://ec.europa.eu/jrc/>

These NE candidates are inspected for type indicators (e.g. if an NE is followed by a company indicator like “Ltd”, “LLC”, “EEIG”, etc. it can be assumed to be a company). If no indicator can be found, the entity is returned as an ‘entity’ of an unknown type.

3.5.3 Coreference

All entities of a text are collected in an NE list. This list is used to assign type information to unspecified entities. Currently, a very simple coreference mechanism is used (string inclusion); more sophisticated approaches will be investigated in later versions.

3.6 Integration into the MULTISENSOR platform and status

NER runs on Linguattec servers. The communication between the NER server and MULTISENSOR platform will be established via REST web services (see above).

A first version of the system for English has been released; access via HTTP has been implemented. Lexicons for 4 languages (English, French, Spanish, and German) have been produced, and grammars to support their lookup are available.

Ongoing tasks are: NIF integration, resolving named entities against DBPedia, and extension of the functionality to cover names containing unknown parts.

4 CONCEPT EXTRACTION FROM TEXT

4.1 Management Summary

This section describes the progress in task T2.3. An annotation exercise is being carried out where texts belonging to the UC1 scenario 1 (energy policies) are being annotated with their mentions to entities potentially found in the datasets of the semantic repository, and also with lexical information obtained from selected lexical resources. The main purpose of the annotation exercise is to evaluate the coverage of the datasets and lexical resources over the information communicated in the texts, on one side and the linguistic constructions used to convey this information, on the other side. The annotation is also useful for the development of automatic methods and future evaluation of the task. For this reason, and given the complexity of the manual annotation, a set of guidelines has been developed.

4.2 Approach

The goal of T2.3 is to identify in the text all explicit mentions to concepts that belong to the project domains and are modelled in the ontologies and datasets. Progress in this task is therefore dependent on the degree of characterisation of the scenarios and the determination of the vocabularies and datasets to be used for each scenario. At the time of planning the work for tasks T2.3 and T2.4, UC1 scenario 1 (energy policies) was the scenario which had advanced the furthest, and various relevant lexical resources and datasets had been proposed. For this reason, our efforts for this task have been so far focused on this scenario, albeit the general procedures are valid for all the use cases.

Concept extraction starts from the results of the NER task (T2.2). The mentions to entities detected by the NER task include references to specific individuals, organisations, locations and time periods. The methods and general approach of the NER implementation are geared towards the detection of general-domain entities, as found in encyclopaedic resources like the Wikipedia¹¹/DBPedia¹², and therefore domain-specific terminology and actors might be missing. The concept extraction task aims at extending the recognition of general-domain entities to include references to domain-specific ones. While an encyclopaedic resource like DBPedia can be used as a target dataset for the detection of entities belonging to general discourse, the detection of domain-specific entities requires vocabularies, ontologies or datasets specific to each scenario.

The core of our approach to concept extraction is to make the most of existing lexical resources in order to effectively identify mentions to entities in the domain datasets. More precisely, we aim at using these resources to collect multilingual lists of terms and expressions which are used in the domain to refer to relevant entities. Lexical databases, however, may not cover completely the terminology and main actors of a scenario, either because of being incomplete, outdated, or not covering an overlapping but not identical domain. The starting point for our approach, therefore, is a manual annotation exercise where domain texts are annotated with the entities in the target datasets that they

¹¹ See <http://www.wikipedia.org/>

¹² See <http://dbpedia.org/>

communicate, and for each mention we write down if it can be found in the lexical resources under consideration. This annotation is extended to cover relations between entities, as explained in C.

The purpose of the annotation exercise is to (i) assess the coverage of lexical resources over the linguistic expressions used in domain texts to refer to relevant entities and relations between them, (ii) to create a development set of annotated texts which can be used as a seed for unsupervised or semi-supervised methods for content extraction (T2.3 and T2.4), and (iii) to eventually create a gold standard for each scenario against which the methods for content extraction developed as part of WP2 can be evaluated. The exercise has focused so far on the annotation of texts in English pertinent to the energy policies scenario. A set of annotation guidelines, which can be found in Appendix: Annotation Guidelines, is being developed in order to guide the annotation and ensure a certain level of agreement between annotators. As described in them, the annotation covers both general-domain and domain-specific entities, corefering entities, and relations indicated by predicative verbs and nouns.

Being a gold standard for the content extraction task, the text analysis pipeline in WP2 aims at producing results as close as possible to those of the manual annotation. Eventually, the content extraction pipeline should be able to automatically produce texts semantically annotated with a reasonable quality. As described in deliverable D6.1 for the summarisation package (WP6), both the manually annotated gold standard and the set of texts analysed by the content extraction pipeline will also constitute a corpus for the development of AS and NLG methods.

4.3 Software

The annotation uses Brat¹³, a web-based tool designed for structured annotation. This tool has been used in the past for the creation of datasets for various Natural Language Processing (NLP) and Information Extraction (IE) tasks. Its support for n-ary relations between annotations and enriching annotations with additional information makes it particularly suited to our annotation requirements. We use an annotation schema where three types of annotations are foreseen: entities, coreferent entities and relations. The first annotation type is used for nominal expressions that introduce new referents in the text; the second is used to mark further mentions, while the third is used to mark predicates which indicate n-ary relations between entities or other relations. Each annotation type has associated a set of attributes which are used during annotation to mark whether the annotated text is found in the lexical resources considered for the annotation, and what other information is found in these resources (e.g. translations to other languages, semantic roles assigned to arguments of predicates). A version of the brat tool configured with this annotation solution has been deployed in a server maintained at UPF¹⁴ and is at the moment being used by three linguists to annotate texts belonging to the energy policies domain.

¹³ See <http://brat.nlplab.org/index.html>

¹⁴ See <http://brat.taln.upf.edu/#/>

4.4 Resources

The choice of lexical resources and target datasets is mostly driven by their inclusion in the list of data sources for the knowledge modelling task (T5.1), described in D5.1. The following lexical resources are being used for the annotation task:

1. BableNet¹⁵, a multilingual index mapping entries from Wordnet¹⁶, Open Multilingual Wordnet¹⁷, Wikipedia, DBPedia, Wiktionary¹⁸ and others.
2. Unified Verb Index (UVI)¹⁹: an English index mapping entries from the verb dictionaries VerbNet²⁰ and the Proposition Bank²¹, and the predicative sense dictionary FrameNet²².
3. Eurovoc²³: multilingual thesaurus containing terms obtained from official documents of the European Parliament. An RDF version of the resource has been recently published as an LOD dataset.
4. Reegle Glossary²⁴: multilingual thesaurus containing terms belonging to the renewable energies domain.

BableNet and UVI aggregate several general-domain resources under a single search interface. The main reason for choosing BableNet is its mapping of multilingual expressions and lexical resources to the large-scale encyclopaedic DBPedia. UVI, on the other hand, is the largest lexical database dedicated to verbs. It is being used in the annotation of verbal predicates. Eurovoc and Reegle are domain-specific resources which focus on the terminology of parliamentary activities in the European Union and renewable energies respectively. They are also the main data sources for the energy policies scenario according to D5.1. Eurovoc and Reegle do not cover exactly the vocabulary of journalistic texts about energy policies. This constitutes a further reason to estimate their actual coverage.

The following LOD datasets are being considered as target datasets:

1. DBPedia: DBPedia is an RDF dataset with data extracted from semi-structured data in Wikipedia pages. Most pages in Wikipedia have their corresponding entry in DBPedia, an entity identified by a dereferencable URI.
2. Reegle data²⁵: an RDF dataset containing entities for the terms in the Reegle Glossary plus a list of important actors (organisations and individuals) for the renewable domain.

All these resources should eventually be used in the automated content extraction pipeline.

¹⁵ See <http://babelnet.org/search.jsp>

¹⁶ See <http://wordnet.princeton.edu/>

¹⁷ See <http://compling.hss.ntu.edu.sg/omw/>

¹⁸ See <https://www.wiktionary.org/>

¹⁹ See <http://verbs.colorado.edu/verb-index/search.php>

²⁰ See <http://verbs.colorado.edu/~mpalmer/projects/verbnet.html>

²¹ See <http://verbs.colorado.edu/~mpalmer/projects/ace.html>

²² See <https://framenet.icsi.berkeley.edu/fndrupal/>

²³ See <http://eurovoc.europa.eu/drupal/>

²⁴ See <http://www.reegle.info/glossary>

²⁵ See <http://www.reegle.info/index.php>

4.5 Evaluation

No formal evaluation has been conducted yet. However an evaluation of the annotation guidelines is foreseen. This evaluation will be based on the annotation of text fragments following the guidelines by two or more linguists, and will use the free-marginal multi-rater Kappa. This metric, used for the evaluation of the Content Selection Challenge 2013 (Bouayad-Agha et al., 2013), measures the agreement between annotators and is suited to annotation tasks with multiple raters and discrete annotation categories. The validation for the guidelines is a pre-requisite for the annotation of a gold standard for content extraction. Its evaluation will be carried out in the weeks following the submission of this deliverable.

4.6 Integration into the MULTISENSOR platform

As indicated in D.7.1, both implementations of concept extraction (T2.3) and relation extraction (T2.4) tasks will be deployed as part of a single content extraction module. The output of this module will consist of RDF triples encoded in JSON-LD²⁶.

²⁶ See <http://json-ld.org/>

5 CONCEPT LINKING AND RELATIONS

5.1 Management Summary

This section describes the progress in task T2.4. The annotation exercise described in the previous section is also relevant for the linking of concepts and the extraction of relations between them. Besides the annotation, work in task T2.4 has focused on the development of a multilingual deep dependency parser. Its development is crucial for the task as the output of the deep parser is the starting point for the extraction of relations.

5.2 Approach

The general approach to the extraction of triplets is to identify common patterns in linguistic predicate-argument structures. These structures can be obtained by applying a deep parser or semantic role labeller (SRL) which, for each nominal or verbal predicate, mark its arguments in the text and assign labels that describe the role or position of the arguments in the predicate structure.

Deep parsing has already been used for IE. In (Draicchio et al., 2013), the deep structures produced by the Boxer parser (Curran et al., 2007, Bos 2008) are mapped to an OWL/RDF representation. Boxer performs not only deep syntactic analysis, but also NER and coreference resolution. The logical representation of natural language text generated by Boxer is transformed into OWL/RDF, using VerbNet and FrameNet semantic roles when possible, and domain specific ones otherwise. The results are further enriched with NED against DBpedia, as well as with WSD against WordNet that allows linking relations to DUL (Dolce + DnS Ultralite) classes and other ontologies on the Web such as Schema.org. A similar approach is followed in LODifier (Augenstein and Rudolph, 2012) that aims to generate Linked Data from unstructured text by using Boxer, NED against DBpedia, and WSD against WordNet. Unlike (Draicchio et al., 2013), LODifier uses blank nodes when transforming Boxer's output into RDF/OWL and also provides a higher extent of interlinking with DBpedia and WordNet, by attempting to link to DBpedia all entity and relation mentions whenever possible, and to their corresponding WordNet synset otherwise, contrary to (Draicchio et al., 2013) that considers mainly NE and relations.. The system presented in (Exner and Nugues, 2013) goes a bit further and uses no new vocabularies but just the DBpedia ontology to represent the extracted data. Rather than a full deep parser, their system uses an SRL tool based on the Proposition Bank to detect predicates and their arguments. NED and Coreference resolution are then applied to map as many of the arguments as possible to entities in DBpedia. Finally, roles in the Proposition Bank between pairs of entities are mapped to binary RDF properties in the DBpedia ontology, resulting in triples expressed using DBpedia entities and vocabulary.

In contrast to the binary relations expressed by ontological properties in datasets like DBpedia and Reegle, predicate-argument structures are n-ary and therefore require greater expressiveness for their representation. However, it is often desirable to reuse domain ontologies rather than using ad-hoc or linguistic vocabularies to encode the results of deep parsing. This opens the door to two alternative approaches for the extraction of relations. Either predicate-argument structures are mapped to multiple binary relations found in existing ontologies, as in (Exner and Nugues, 2013), or the n-ary relations expressed by

linguistic predicates are modelled explicitly. We aim at exploring both approaches, and in particular we are considering using FrameNet as a repository of language-independent predicative senses to represent n-ary relations.

5.3 Software

The deep parser (Ballesteros and Bohnet, 2014) being developed and trained for WP6 delivers deep-syntactic dependency structures from sentences in natural language. This type of dependency structures, to which we refer to as DSyntS, captures the argumentative, attributive and coordinative relations between full words of a sentence, while abstracting away functional aspects of the analysis of sentences. Their abstraction degree falls somewhere between the output of a surface syntactic dependency parser (SSyntS) which consists of connected trees defined over all words of a sentence and language-specific grammatical functions, and the output of a semantic parser, the latter being forests of trees defined over individual lexemes or phrasal chunks, and abstract semantic role labels which capture the argument structure of predicative elements, dropping all attributive and coordinative dependencies. Figure 3 shows (a) an SSyntS of an English sentence and (b) a DSyntS of the same sentence:

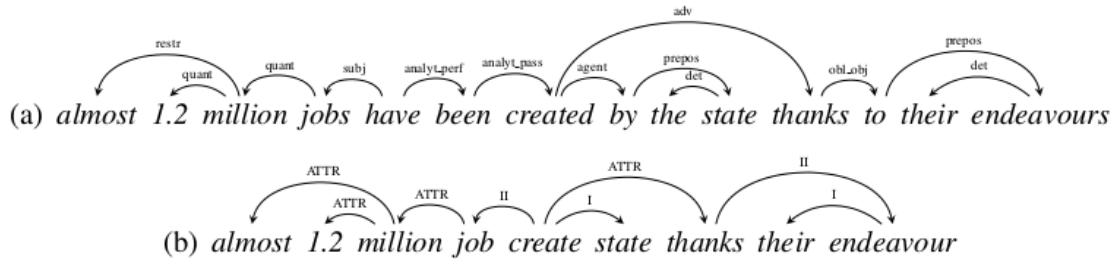


Figure 3: SSyntS and DSyntS structures of an English sentence.

The parser works in a pipeline and makes use of a joint transition-based tagger (Bohnet and Nivre, 2012) that delivers the surface syntactic structures which constitute the input of the deep-syntactic parser, also known as SSynt-Dsynt transducer. Figure 4 illustrates the pipeline:

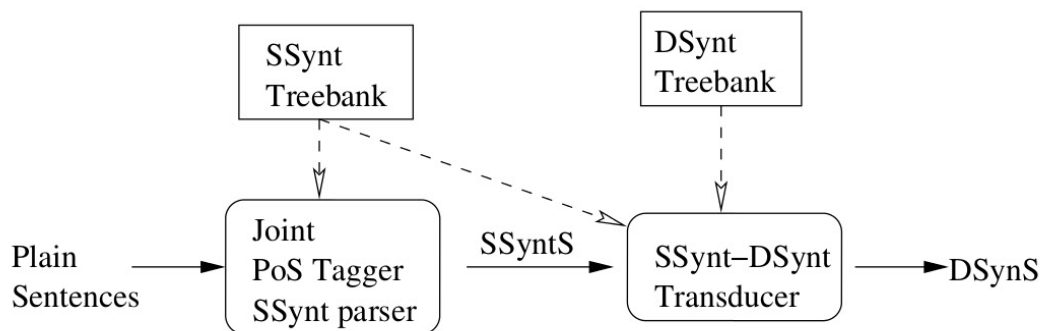


Figure 4: deep dependency parser pipeline.

5.4 Training resources

In order to train the pipeline, it is necessary to use multilingual, parallel morphologic, surface and deep-syntactic treebanks (Mille et al., 2013), that is a corpus for each language where

each sentence is manually annotated with its morphological, SSynt and DSynt structures. We train language-specific versions of the deep parser using the same datasets also used to train the surface generation module of the abstractive summarisation task, described in D6.1, Section 3.2. The joint transition-based tagger is trained on the morphologic and surface-syntactic layers together with the input text, while the SSynt-DSynt transducer is trained on the surface- and deep-syntactic layers.

5.5 Integration into the MULTISENSOR platform

The deep dependency parser is deployed in its own service, given that it will be used for other tasks (i.e. sentiment analysis in WP3). Its output is encoded using the NIF 2.0 ontology²⁷ for stand-off annotation, and serialised using the JSON serialisation format for RDF linked data JSON-LD. At the moment, the output includes the deep analysis, but it will be extended to also include the results of the surface analysis for each sentence.

Both concept extraction (T2.3) and relation extraction (T2.4) will be deployed in a single service that will perform general content extraction. The results of the content extraction module will consist of RDF data serialised using JSON-LD. These data will be ready to be asserted in the semantic repository.

²⁷ See <http://persistence.uni-leipzig.org/nlp2rdf/>

6 AUDIO TRANSCRIPTION

6.1 Management summary

Automatic speech recognition (ASR) is employed within the MULTISENSOR project to provide a channel for analysis of spoken language in audio and video files. The transcripts produced by ASR follow the same analysis and indexing procedure as the input from other text sources. The languages covered by the ASR component are English and German.

6.2 State of the art

Automatic speech recognition is a transformation of speech signal to a sequence of phonemes and words. The recognition quality depends on different factors such as speaker and channel variability, background noises, audio frequency spectrum, quality of microphone, or difficulty in differentiation between speech and non-speech events. State-of-the-art systems apply different methods in acoustic and language modelling, feature extraction, and decoding, for handling these issues and overcoming the problems. Most systems apply statistical approaches, typically based on HMM (cf. Gales and Young, 2007).

ASR systems can be differentiated according to several criteria:

- Small or large vocabulary: systems able to manage (a) a very small number of words, e.g. only numbers, a few commands, or a very limited narrow-domain vocabulary (b) large vocabularies with up to several hundreds of words for dictation tasks or similar;
- Discrete or continuous recognition: systems where (a) discrete recognition requires the speaker to pause between words (b) continuous recognition allows speakers to use their normal speech flow, running words together without any pauses between them;
- Speaker-dependent or independent: (a) speaker-dependent systems expect the user to train the system, to make it more sensitive to his voice and his usual background environment (b) speaker-independent systems do not require any training
- Desktop or server-based application: (a) systems requiring local software installation on computers where ASR is used or (b) system that are centralised in a server and can be accessed from different remote clients

Today, most ASR systems belong to the LVCSR (Large Vocabulary Continuous Speech Recognition) systems.

6.3 Technical framework and approach

Several open source systems are available, such as HTK Toolkit²⁸, Sphinx-4²⁹, Julius³⁰, RWTH-ASR³¹, and Kaldi³².

²⁸ <http://htk.eng.cam.ac.uk/>

²⁹ <http://cmusphinx.sourceforge.net/>

³⁰ http://julius.sourceforge.jp/en_index.php

³¹ <http://www-i6.informatik.rwth-aachen.de/rwth-asr/>

³² <http://kaldi.sourceforge.net/>

In MULTISENSOR, the speech recognition is implemented by using the RWTH-ASR (cf. Rybach et al., 2009) toolkit and technical framework (available under an open source license).

This ASR technology is a speaker-independent, server-based, LVCSR (cf. Ney et al., 1998), which also allows using the open-vocabulary approach (recognition of unknown words based on sub-word units). It employs a series of state-of-the-art techniques: continuous density HMMs for the acoustic modelling; MFCC or PLP feature extraction with support of LDA and VTLN; speaker adaptation by the means of CMLLR; time-synchronous left-to-right beam search strategy for the decoding. By this means, the toolkit provides the basic modules for signal analysis and feature extraction, acoustic modelling and decoding. It does not include tools for estimation of language models itself, but it supports N-gram language models in ARPA format, as well as class language models, or weighted grammars in form of finite state automata.

In the project we will use the existing acoustic models. They have been trained on approx. 90 hours of news recordings in German and English. As for language models and recognition lexicons, the baseline systems have been developed from scratch, trained on freely available data. For language modelling, we use SRILM³³ tool for LM estimation.

The advanced versions will be adapted by using in-domain data, as much as the project partners manage to collect. Another issue in the development of advanced systems will be the integration of the recognition results of the named entities recogniser in the last phase of the project. This should ensure better recognition of proper names in spoken language.

6.4 Implementation and components

During the first year of the project, the main focus was on recognition lexicons, language models and required pre- and post-processing components.

Nowadays, most ASR systems can deal with large vocabularies and huge language models. However, they mostly use a closed vocabulary, so that the recognition is static and limited only to those words defined there. Our system follows the open-vocabulary approach (cf. Hahn & Rybach, 2011) and is far more flexible; it allows the recognition of OOV words compounded from word fragments defined in the lexicon and trained in the language model. The recognition lexicons and language models are hybrids between whole words and sub-words, such as compound parts, morphemes, prefixes, suffixes etc. The process of their creation involves a series of statistical and linguistic modules, which will be described in the following: (1) data preparation: corpora collection and pre-processing (2) pronunciation dictionary: vocabulary selection and dictionary creation (3) language model estimation.

6.4.1 Data preparation

The baseline LMs for German and English are based on many open source text resources, as well as data crawled from the web. The largest freely available corpora belong to the European Parliament speeches³⁴, DGT's translation memories³⁵, and the JRC-ACQUIS

³³ <http://www.speech.sri.com/projects/srilm/>

³⁴ <http://www.statmt.org/europarl/>

³⁵ <https://ec.europa.eu/jrc/en/language-technologies/dgt-translation-memory>

corpus³⁶. Being multilingual parallel corpora, they have primarily been collected for the machine translation task, but the monolingual parts of them build a solid basis for language modelling for ASR as well. Furthermore, we have obtained access to several newspapers and news agencies archives (Süddeutsche Zeitung, TAZ, Die Zeit, and Reuters). The data crawled from web include different topics, such as occupational health and safety, and environment.

Crawler and HTML Parser

For the purposes of monolingual crawling, we adapted a bilingual crawler developed by Linguattec (Kaumanns, 2011). The crawler uses seed URLs but is not a focused crawler.

HTML-files are stored and then offline processed by a special HTML-Parser written at Linguattec (cf. Kaumanns, 2011) that provides the following functionality: separates text from boilerplates; identifies paragraph types (heading, text, enumeration, table cell); removes inline tags (formatting, href); removes corrupt text portions (e.g. containing more non-letters than letters etc.); removes intros (text ending in '...') etc.

Output of the component are documents in an internal Linguattec format called LTBasic. This format is the input for all other data preparation steps (sentence segmentation, tokenisation etc.).

Sentence splitter

The texts have to be first divided into sentences, which are the standard unit in language processing, and the basis of all following analysis tasks.

The Linguattec tools detect sentence boundaries by applying rules about how to process punctuations, end-of-sentence words, start-of-sentence words, different kinds of abbreviations (e.g.: 'Prof.' is never sentence-final, while 'etc.' is very often sentence-final), etc.

Tokeniser

The role of the tokenisation component in this workflow is to detach punctuation marks, brackets, symbols (% , \$, §) etc. from words. The detached punctuation marks (commas, periods, colons, as well as brackets etc.) can undergo two different procedures:

- they should be removed entirely from the corpus, if an ASR system without punctuation commands is being built
- or, for systems where the users explicitly speak the punctuation, they are kept and undergo further pre-processing steps (see below)

Other detached symbols (currencies, percent marks, measures etc.) are being kept and go to the normalisation step.

Normaliser

The normalisation step comprises following tasks:

- Variants of the same token are harmonised, e.g.: Dr., dr., Dr, Doctor => doctor
- Numbers, dates, currencies, symbols etc. are spelled, e.g.: Oct. 1st => first of October
- Punctuations are converted into special tokens: , \COMMA, : \COLON, . \PERIOD etc.
- Abbreviations, acronyms, symbols are expanded: etc. => et cetera, § => paragraph

³⁶ <https://ec.europa.eu/jrc/en/language-technologies/jrc-acquis>

Decomposer

Originally, this software was developed by Linguattec for decomposing German compounds into their parts. For the open-vocabulary ASR approach, we adapted it slightly to produce a kind of morphological segmentation of English words as well.

Compound words found in the training corpus, but not in the base recognition lexicon (see below), are segmented into smaller units which are then added to the lexicon as sub-words, according to their corpus frequency. If they are able of building a modifying part of a compound, they are marked by a plus-sign (kinder+). Doing so, we enable the system to recognise many possible combinations of kinder+ plus different heads (wagen, garten, zimmer) dynamically, instead of entering a very huge, but nevertheless limited, number of compounds into the lexicon (kinderwagen, kindergarten, kinderzimmer).

In the end, all training data have to be lower-cased, and each sentence is marked by a sentence-start and a sentence-end symbol. This is the input for the language modelling tool.

6.4.2 Language models

The goal of language modelling for ASR is assigning of a probability to a sentence or to a sequence of words, i.e. to a recognition hypothesis. Having acoustic evidence and one or many hypotheses, the job of a language model is to assign higher probability to frequent observed sequences of words and to penalise the rarely seen ones (the probability of “Eye like two right on my pea sea without mist aches” might be lower than that of the sentence “I like to write on my PC without mistakes”).

We build probabilistic N-gram language models, which are able to effectively validate word sequences and assign higher probabilities to most plausible ones. There are several tools for language modelling, such as SRILM (Stolcke, 2002), IRSTLM (Federico et al., 2008), KenLM (Heafield, 2010) etc. For the language model estimation we use SRILM; as the final LM a binary file is created by the ASR tool and deployed, for faster loading.

For the German baseline, the training corpus consisted of ca. 390 million sentences (resulting in an ARPA model of ca. 24 GB; binarised ca. 6 GB). The English data contained ca. 15 million sentences (ARPA model ca. 1.5 GB; binarised ca. 500 MB). The sizes of the corresponding vocabularies are: ca. 600k entries in German and ca. 150k entries in English. This big difference in sizes is not only a result of the data availability, but it is intentional as it helps us to best answer the question of how the size of the resources correlates to the recognition quality.

6.4.3 Pronunciation dictionary

The pronunciation lexicon contains a selection of words (and word fragments) that occur in the training corpus. The lexicon has three main sections:

- A list of all phonemes in the respective language
- A list of special lemmas such as noise, silence, sentence begin, sentence end, hesitation, and unknown word (if used, all of them need to be trained in the acoustic model and/or language model)
- A list of words and word parts for the recognition; each entry contains as obligatory information a lemma and one or several pronunciation variants.

Vocabulary selection

The task consists in selecting the list of words to populate the recognition lexicon.

First, we include into the selection all function words which usually constitute the basis of every lexicon: articles, pronouns, conjunctions, modal and auxiliary verbs, prepositions, non-derivative adverbs etc. This is based on existing Linguatéc lexicons.

The selection of all other words is corpus-based and follows primarily the corpus frequency. The most frequent ones are included into the lexicon as whole words. The rest is decomposed, and the most frequent fragments are selected. It is up to the lexicon developer to decide how many entries will be included into the lexicon. The more entries the better chances to lower the OOV rate during the recognition. But, since big vocabularies can slow down the recognition process significantly, it is a matter of running experiments to find the best ratio vocabulary size/recognition performance.

Pronunciation assignment

Each lexicon entry requires at least one phonetic representation. The pronunciation assignment is done automatically, from a big database of already generated word/pronunciation pairs. If a word is not in the database, we use a statistical G2P (grapheme to phoneme) model to generate a default pronunciation. For the training, we use the open-source software Sequitur G2P³⁷.

A special problem is the pronunciation of foreign words and names which requires more manual effort and quality checks.

6.5 Evaluation

In ASR, the recognition quality is measured by WER (word error rate). For this, a reference set of manually transcribed documents is needed.

In MULTISENSOR, no formal evaluation has been conducted yet. In the case that the manual creation of the reference set is not affordable for both languages (1 hour audio requires 30-60 hours manual transcription work), we will use the existing test set for German, created by Linguatéc for internal purposes, and will only have to create a test corpus for English.

6.6 Integration into the MULTISENSOR platform

The speech recognition system runs on Linguatéc servers. The communication between the ASR server and MULTISENSOR platform is established via REST web services.

³⁷ <http://www-i6.informatik.rwth-aachen.de/web/Software/g2p.html>

7 TECHNIQUES FOR MULTIMEDIA CONCEPT DETECTION

This section presents the techniques applied in multimedia concept detection, which involves the detection of a set of predefined concepts in multimedia files including videos and images, by considering visual and textual features. Visual features are extracted in the current component, while textual features are derived from the speech recognition. In the current deliverable, we shall focus exclusively on the visual features while the fusion with the textual features will be presented in the next deliverable (D2.3).

Multimedia concept detection involves the following steps:

- Video decoding: this step is applied only in case that the input file is a video and it is responsible for extracting specific frames from the video.
- Feature extraction: this step refers to the extraction of descriptors that describe visually the images/ frames by capturing global or local information.
- Classification: this step is related to the development of models used for classifying images or video frames to the set of predefined concepts/ categories.

7.1 State of the Art in Multimedia concept detection

This section presents a brief overview of the state-of-the-art techniques and methods of the aforementioned three domains (i.e. video decoding, feature extraction, and classification).

7.1.1 Video decoding

As we mentioned already, video decoding is applied when the input file is a video and it involves the extraction of frames out of the video that will be used for representing the video (i.e. keyframes). Here, we will present only techniques related to the temporal segmentation of a video into elementary units, while other types of decoding will not be covered (e.g. spatiotemporal segmentation). Specifically, three approaches are described for temporal segmentation and selection of the video keyframes.

Segmentation with a predefined time-step

In this approach, we consider as keyframes the video frames retrieved with specific time rate. The time rate is selected almost randomly by the researcher by solely considering the desired number of keyframes for representing the video. Therefore, the total number of keyframes for a specific video can be easily estimated, since the only prerequisites are the video duration and its frame rate. This approach is rather straightforward, since it does not involve any processing and consequently, however since no analysis is realised, we cannot guarantee the quality of keyframes for capturing the essence of the video.

Shot segmentation

The second approach aims at partitioning the video into consecutive frames, called shots. Since shots are defined as continuous temporal segments, shot segmentation can be handled as detection of the video shot boundaries, i.e. the temporal limits of each shot. Then, one or more frames from each shot (i.e. keyframes), are used for representing the shots. The first approaches proposed for shot segmentation, performed shot boundary detection based on pair-wise pixel comparisons between successive or distant frames of the video stream. Later, more sophisticated approaches were proposed that considered

different type of structural features (e.g. color, edge) or different visual features (Tan et al., 2007; Liu et al., 2008). Also, more complex approaches involving the use of Support Vector Machine (SVM) classifiers were proposed (Ling et al., 2008; Tsamoura et al., 2008). Moreover, techniques that perform shot segmentation without applying a prior video decompression into frames were proposed (Doulaverakis et al., 2004). A detailed overview of shot segmentation techniques can be found in (Apostolidis et al., 2012; Stein et al., 2013).

Scene segmentation

Scenes, compared to shots, are higher-level temporal segments covering a single event or several related events taking place in parallel. The segmentation of a video to scenes allows the organisation of its content in higher levels of abstraction. For keyframe extraction, one or more frames are selected from the scenes. In general, scene segmentation techniques use as input the shot segments and try to group them according to their semantic similarity. As far as the scene segmentation techniques that consider only visual information are concerned, four main categories can be recognised: (a) graph-based methods (Yeung et al., 1998); (b) methods using inter-shot similarity measurements (Rasheed and Shah, 2003); (c) clustering-based methods (Liao and Zhang, 2007); and (d) other methods (Zhai and Shah, 2006; Zhu and Liu, 2009). Finally, there is a group of techniques that considers audio and visual information (Sidiropoulos et al., 2011). A detailed overview of scene segmentation techniques can be found in (Apostolidis et al., 2012).

7.1.2 Feature extraction

The feature extraction step involves the methods that aim at the effective description of the visual content of images. In general, many descriptors have been introduced for the representation of various image features and can be divided into two main groups, the global and the local descriptors. The difference between these categories is the locality of the feature that is represented. Specifically, global descriptors use global characteristics of the image, while local descriptors represent local salient points or regions. Usually, when local descriptors are employed, a clustering algorithm is applied to form a vocabulary of “visual words”. Such representation is required, since a point-to-point comparison in content-based image retrieval applications is a time and CPU-processing demanding procedure. In the end, a global descriptor is produced that gives an overall impression of visual data. In the following subsections we present briefly some descriptors from the relevant literature.

Global descriptors

The algorithms for capturing the global characteristics of images can be distinguished into two categories; the first includes the ones that describe features at image-level, while the second includes those that attempt to capture arbitrarily shaped regions within the image. Regarding the first category, a number of different techniques were developed that capture the images’ characteristics, such as color and texture, using for example the MPEG-7 descriptors, the Grid Color Moments, the Gabor Texture, and others. Some examples of works using such descriptors are the following (Huiskes et al., 2010; Hauptmann et al., 2006). A significant amount of works were realised in relation to the second category as well. Some propose the segmentation into regions and then the extraction of color and texture information (Souvannavong et al., 2005); while others combine color and texture descriptors with region-based shape descriptors (Chang et al., 2007).

Local Descriptors

The descriptors that fall within this category are locally extracted. The first step of the algorithms used is the definition of the points of interest in the image, and the second is the extraction of visual descriptions for each point. Regarding the identification of the interest points, several techniques have been proposed, such as Harris-Laplace point detector (Tuytelaars and Mikolajczyk, 2008); dense sampling strategy (Jurie and Triggs, 2005); and others. Regarding the visual representation of the interest points, several descriptors have been proposed. The most broadly used is the SIFT descriptor (Lowe, 2004) and its extensions, HSV-SIFT (Bosch et al., 2008), HUE-SIFT (van de Weijer et al., 2006), OpponentSIFT, rgSIFT, C-SIFT, and RGB-SIFT (Van De Sande et al., 2010). Another local feature descriptor, based on SIFT, is the SURF descriptor (Bay et al., 2008), which also has several variations such as dense-SURF (Tao, 2011). Finally, several attempts were made that combined global and local descriptors (Jiang et al., 2008; Chang et al., 2007).

Visual Word Assignment

In the visual word assignment step, local descriptors are transformed to a “bag-of-words” (BoW) representation (Qiu, 2002). Similar keywords are grouped in clusters and each cluster is treated as a visual word that forms the visual vocabulary. Then, the local descriptors are assigned to this vocabulary in a manner that each descriptor is mapped to a visual word. The method that is usually applied for constructing the visual vocabularies is the K-Means clustering algorithm, while next the Nearest Neighbor algorithm is applied for assigning descriptors to visual words (Zhang et al., 2007; Van De Sande et al., 2008). It should be noted that although the BoW approach is the most popular encoding, other approaches have been proposed that seem to outperform it according to several studies (Van de Sande et al., 2014; Chatfield et al., 2011). These approaches are the Fisher vector (FV) described in (Perronnin et al., 2010), and the VLAD (Vector of Locally Aggregated Descriptors) described in (Jegou et al., 2010). VLAD is a fast approximation of FV that is slightly inferior in terms of performance but is more compact and fast to compute (Jegou et al., 2012).

7.1.3 Classification

The classification step is the last step of the multimedia concept detection procedure. It develops models for concept detection by using the low-level visual features, and then it performs image labelling. Thus, classification aims at the automatic understanding of the image/ video by looking at the visual content. However, in the advanced version of multimedia concept detection, textual information will be considered as well.

In the last years, there has been a significant effort towards discovering techniques that allow the efficient video concept detection. Significant boost was given by the TRECVID Semantic Indexing task (Smeaton et al., 2009) that targeted handling large amounts of video data and detecting multiple semantic concepts (e.g. (Wei et al., 2012; Snoek et al., 2011)). For learning the associations between the image representations and concept labels, algorithms such as Support Vector Machines (SVM) are trained separately for each concept, on ground-truth annotated corpora. Then, when a new unlabeled video shot arrives, the trained concept detectors will return confidence scores that show the belief of each detector that the corresponding concept appears in the shot. For each descriptor used, a new classifier is trained and in the end, fusion of the values is realised. More recent approaches

proposed considering correlations instead of training each SVM independently (Markatopoulou et al., 2014).

7.2 Approach

In this section, we present the techniques that were applied in MULTISENSOR as part of the basic version of multimedia concept detection. Figure 6 depicts an overview of the multimedia concept detection procedure.

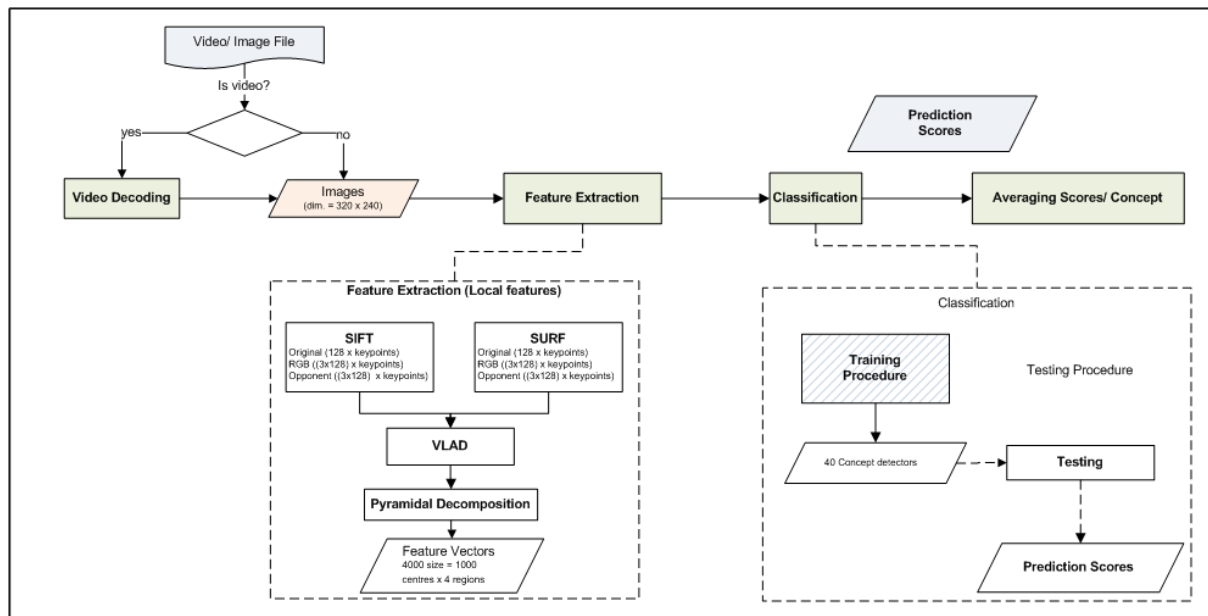


Figure 6: Overview of the multimedia concept detection procedure.

7.2.1 Video decoding

Segmentation with a predefined time-step

Although this approach is the most straightforward and simple when attempting to represent a video with frames, it should be noted that in the current deliverable, it was evaluated only in terms of time, and not used for the video representation to keyframes.

Shot segmentation

The employed technique used for the decomposition of the video into shots is based on the algorithm presented in (Tsamoura et al., 2008) and its extension presented in (Stein et al., 2013). The applied algorithm takes as input a video and detects transitions based on global and local visual information. Then, feature vectors are created for each frame and the distances between neighboring frames are computed, forming the corresponding distance vectors. The evaluation of these vectors is realised using SVM classifiers, which identify the shot boundary between each pair of consecutive frames. Finally, by using pre-defined decision rules and temporal constraints, the frames corresponding to camera flash-lights are detected, and the false shot boundaries are discarded. This algorithm was tested on the LinkedTV³⁸ News Show and the Documentary scenarios and performed remarkably well. A

³⁸ LinkedTv EU project, <http://www.linkedtv.eu/>

detailed presentation of the evaluation can be found in (Stein et al., 2012a; Stein et al., 2012b).

Scene Segmentation

The technique used for the segmentation of videos to scenes, is the segmentation algorithm proposed by (Sidiropoulos et al., 2011). This technique groups shots into sets, based on the content similarity and temporal consistency among shots. Content similarity in the original algorithm considers information from different modalities. However, the version of the algorithm applied here, which was used also in (Apostolidis et al., 2014) considers only the low-level visual information for checking the content similarity. This is realised mainly for time-processing reasons, since although the high-level concepts improve the performance of the algorithm, they introduce significant computation complexity and time delay.

The experiments presented in (Sidiropoulos et al., 2011), proved that the applied technique is capable of identifying the scene level structure of videos rather well, while the processing time is a small fraction ($< 1\%$) of the overall analysis. It should be noted that in the current deliverable, the scene segmentation technique described is evaluated only in terms of time, and it is not used for video representation to keyframes.

7.2.2 Feature extraction

Regarding the features that were used in the first version in the multimedia concept detection, they are strictly local features. Specifically, we use the latest, most widely used descriptors in concept detection applications, that is the SIFT descriptor and its variations (i.e. RGB-SIFT and opponent-SIFT) and SURF descriptor and its variations (i.e. RGB-SURF and opponent-SURF). Visual words assignment is realised using VLAD encoding. Regarding the dimensionality reduction algorithm that was used, it is PCA and the approach followed is described in (Markatopoulou et al., 2015).

As far as global descriptors such as MPEG-7 and HSV histograms are concerned, we plan on incorporating them into the second version of the system.

7.2.3 Classification

The classification algorithm that was used for learning the associations between the image representations and concept labels was Support Vector Machines (SVM). SVMs are typically used in such cases with most common that of the Semantic Indexing task of TRECVID workshop (Moumtzidou et al., 2010; 2011). In general, SVMs are successfully applied to a significant number of applications in image classification. They are based on the concept of decision planes that define decision boundaries and separates between a set of objects having different class memberships. Therefore, SVM performs classification by constructing hyperplanes in a multidimensional space that separates cases of different class labels.

7.3 Modules description

According to the deliverable D7.1 ("Roadmap towards the implementation of MULTISENSOR platform"), the multimedia concept detection is a single module that incorporates all the aforementioned procedures (i.e. video decoding, feature extraction and classification), receives as input a multimedia file (i.e. image or video) and computes a degree of confidence for each concept. Along the multimedia concept detection procedure a set of tools and

libraries were used. These libraries are either used as they are, that is without any changes, or they have been slightly changed in order to cover our needs.

In the following, we present the tools / libraries used, as well as the input required and output produced in each step of the multimedia concept detection procedure.

7.3.1 Video decoding

Regarding the video decoding techniques, we will focus on the shot segmentation technique, since the system in terms of accuracy was evaluated using only this method.

Shot segmentation

As we have already mentioned the shot segmentation step involves the partitioning of the video into consecutive frames, called shots and the extraction of the keyframes from each shot for representing it. It is therefore evident, that the segmentation step can be analysed into sub-steps:

1. Retrieval of the core elements of the video, called frames. The tool that is used for extracting the video's frames is the ffmpeg library³⁹.
2. Grouping of frames extracted from the previous sub-step into shots using the approach that was earlier described by (Tsamoura et al., 2008). The output is a text file containing the start and end frame of each shot.
3. Selection of one or more frames from each shot for representing the specific shot. These frames are called keyframes. Currently, only one frame is used for representing the shot and it corresponds to the middle frame of the shot.

Therefore, the input of the shot segmentation step is a video file and the output a set of keyframes that represent the video.

7.3.2 Feature extraction

As we have already mentioned, in the current version of multimedia concept detection, we extract only the local descriptors SIFT and SURF and their variations (i.e. RGB-SIFT, opponent-SIFT, RGB-SURF and opponent-SURF). However, before proceeding with the feature extraction, it is necessary to scale all images to the same size. The size that is selected is 320x240 pixels, which is big enough to hold the image information and small enough to allow the concept detection procedure to be time-efficient.

Regarding the feature extraction, the libraries that are used for extracting these descriptors are the following:

- vlFeat⁴⁰, for extracting SIFT descriptor and its variations. vlFeat (Vedaldi and Fulkerson, 2010) implements popular computer vision algorithms.
- OpenCV⁴¹, for extracting SURF descriptor and its variations. OpenCV (Bradski, 2000), is a library of programming functions mainly aimed at real-time computer vision.

³⁹ <http://ffmpeg.org/>

⁴⁰ <http://www.vlfeat.org/>

⁴¹ <http://code.opencv.org/projects/opencv>

The vectors of the descriptors created, after feature extraction, are the following:

- for the simple SIFT and SURF (grayscale) descriptors, vectors with dimensions $128 \times \langle \text{Number of keypoints} \rangle$ were created, and
- for the RGB-SIFT, opponent-SIFT, RGB-SURF and opponent-SURF (color-based), descriptors, vectors with dimensions $3 \times 128 \times \langle \text{Number of keypoints} \rangle$ were created.

Then, all the local descriptors are compacted to 80 dimensions for SIFT, SURF, and their variations using PCA and are aggregated using the VLAD encoding. The result of the above process was a VLAD vector of 163840 elements for SIFT or SURF. Eventually, the VLAD vectors are compressed into 4000-element vectors by applying a modification of the random projection matrix. These reduced VLAD vectors serve as input to the classification step. Figure 7 depicts a part of file with VLAD vectors. It should be noted that each line corresponds to a different image. Such files are used as input both for the classification training and testing phases.

```
1 1: 0.100393 2: 0.056874 3: 0.038531 4: 0.021388 5: -0.035567 ... 3995: -0.028196 3996: 0.036622 3997: 0.075164 3998: 0.004302 3999: 0.044474 4000: 0.015013
1 1: 0.060333 2: 0.041115 3: 0.083851 4: 0.002005 5: -0.060194 ... 3995: -0.049122 3996: 0.055357 3997: 0.068077 3998: -0.004300 3999: 0.054409 4000: -0.050773
1 1: 0.075856 2: 0.017532 3: 0.083756 4: 0.005822 5: -0.036352 ... 3995: -0.024631 3996: 0.037144 3997: 0.021160 3998: -0.020336 3999: -0.011312 4000: -0.024739
1 1: 0.095763 2: 0.072269 3: 0.032646 4: 0.030053 5: -0.053952 ... 3995: -0.039550 3996: 0.041584 3997: 0.003233 3998: -0.053525 3999: -0.041347 4000: -0.008887
1 1: 0.060548 2: 0.094302 3: 0.074458 4: 0.025164 5: -0.057042 ... 3995: -0.050326 3996: 0.064119 3997: 0.073502 3998: -0.055452 3999: 0.097142 4000: -0.030807
.
.
.
1 1: -0.003594 2: 0.044645 3: 0.069197 4: 0.004794 5: -0.059912 ... 3995: -0.053943 3996: 0.033458 3997: 0.024913 3998: 0.010228 3999: 0.081497 4000: -0.044914
1 1: 0.027701 2: 0.084575 3: 0.113921 4: 0.067591 5: -0.003520 ... 3995: -0.035294 3996: 0.040583 3997: -0.002862 3998: -0.015995 3999: 0.055431 4000: 0.004443
```

Figure 7: Part of a file with VLAD vector.

7.3.3 Classification

Finally, as far as the method used for classification (i.e. training and testing) is concerned, it is the Support Vector Machines (SVM). Both for training and testing phases, the classifier receives as input a file similar to the one depicted in Figure 7 and either a new model is developed or this file is tested against the model accordingly. The tool that is used for classification is the LIBLINEAR (Fan et al., 2008) library which is used for the linear classification of large data. Given that we have extracted six descriptors, six models are created for each concept. Then the prediction results for each descriptor per concept are fused using late fusion and specifically by averaging all classifiers output scores. Figure 8 depicts part of the output file.

```
1 1 7.3832e-10
1 2 5.333e-10
1 3 1.9278e-09
1 4 0.0068433
1 5 2.6466e-08
.
.
.
1 56 6.8628e-08
1 57 1.0179e-06
```

Figure 8: Part of an output classification file.

7.4 Concept selection for MULTISENSOR use cases

According to the deliverable D8.2 (“User requirements, specification of pilot use cases and validation plan”), which describes the user requirements for MULTISENSOR use cases, multimedia concept detection is applicable only for the following two use cases:

- Journalism use case scenario
- Commercial media monitoring use case scenario

The users in the aforementioned scenarios need to be able to handle and understand multimedia data including images and videos, while in the third use case scenario (i.e. SME internationalisation use case) such data are absent. Thus, it is necessary to define the concepts that will be handled for each scenario. Moreover, given that these scenarios differ significantly the list of concepts that will be handled for each use case will be different.

The concepts that will be handled are selected with the two following ways:

1. Visual inspection of a set of exemplary videos/ images provided by the user partners
2. Explicit definition by user partners (e.g. logos of specific companies, specific people).

Regarding the concepts selected by the first method, we use two basic sources as concept lists, apart from the ones defined by the technical partners after the visual inspection:

- TRECVID Semantic Indexing task concepts⁴²
- Concepts from the domain-specific ontologies as defined in deliverable D5.1 (Basic Semantic Infrastructure). For the “Journalism use case”, we used the Reegle glossary⁴³, which is used for capturing concepts related to the energy domain, while for the “Commercial media monitoring use case”, we used the DEHEMS ontology⁴⁴ that captures home appliances.

In the sequel, we present the concepts selected for each use case from the two aforementioned sources and the ones that will be addressed in the current deliverable.

7.4.1 Journalism use case scenario

The concepts that are recognised for the journalistic use case are in total 56. Table 1 contains these concepts which are organised into 5 categories based on their source.

Concepts from the Reegle glossary:	TRECVID Concepts:	
• concentrated photovoltaics • flat plate collectors • geothermal power plants • lattice towers • marine current power turbines • photovoltaic power plants • smog	• airplane • anchorperson • bicycle • boat • building • car • cityscape	• factory worker • forest • government leader • explosion/ fire • laboratory • landscape • press conference

⁴² http://www-nlpir.nist.gov/projects/tv2012/tv11.sin.500.concepts_ann_v2.xls

⁴³ <http://www.reegle.info/index.php>

⁴⁴ <http://www.dehems.eu/cms/wp-content/uploads/2011/04/D8.1-Paper3.pdf>

• solar panels • solar power towers • thermal power stations • trains • wind turbines	• demonstration_or_protest • factory • reporters • scientist • waterscape
Concepts found both in the Reegle glossary and in TRECVID:	
• vehicles • aircraft • forestry	
Concepts identified by the technical partners during the visual inspection of the data provided by the user partners:	Concepts proposed explicitly by the user partners:
• banner • car engine • car interior • dam • farm • ice • interview • nuclear energy logo • nuclear reactors • parliament • people • power plant • recycle bin • truck • TV show	• Angela Merkel - Chancellor • Sigmar Gabriel - Minister for Economics • Barbara Hendricks - Minister of BMUB⁴⁵ • Peter Altmaier - former Minister of BMUB • Rainer Brüderle - former Minister for Economics • Norbert Röttgen - former Minister of BMUB • Philipp Rösler - former Minister for Economics • RWE logo (German Energy company) • Vattenfall logo (German Energy company) • E-On logo (German Energy company) • EnBW logo (German Energy company)

Table 1: Concepts for the “Journalism use case”.

The concepts that will be handled in the current deliverable, can be found in Table 2, can also be organised in two categories. The first contains the concepts defined in TRECVID, while the second refers to the concepts recognised by the technical partners or explicitly requested by the user partners.

New concepts		Concepts selected from the TRECVID list	
• solar panels • wind turbines • lattice towers • RWE logo • Vattenfall logo	• E-On logo • EnBW logo • nuclear energy logo • recycle bin • smog	• airplane • boat_ship • car • demonstration_or_protest • explosion/ fire	• forest • government_leader • landscape • press conference • reporters

Table 2: Selected concepts for the “Journalism use case”.

⁴⁵ Ministry of Environment, Nature Conservation, Building and Nuclear Safety (BMUB)

7.4.2 Commercial media monitoring use case scenario

The concepts that are recognised for the home appliance use case are in total 42. Table 3 contains these concepts which are organised into 2 categories based on their source.

Concepts from the DEHEMS ontology:		Concepts proposed explicitly by the user partners:	
• Washing Device • Dishwasher • Laundry Appliance • FabricIron • Clothes Dryer • Tumble Dryer • Clothes Washing Machine • Clothes Washer Dryer • Vacuum Cleaner • Electric Oven	• Stove • Microwave • Freezer • Refrigerator • Fridge Freezer • Breadmaker • Coffee Maker • Food Blender • Food Processor • Electric Kettle • Toaster	• AEG logo • Electrolux logo • Bauknecht logo • BEKO logo • BOSCH logo • Philips logo • Siemens logo • Whirlpool logo • Fagor logo • Gaggenau logo	• General Electric logo • Hoover logo • Indesit logo • LG logo • Miele logo • NEFF logo • Samsung logo • V Zug logo • Zanussi logo • Bissell logo • Dyson logo

Table 3: Concepts for the “Commercial media monitoring use case”.

In the current deliverable we targeted only part of the concepts that were selected from the list with the concepts proposed by the user partners, which contains logos of home appliances companies. The number of the selected concepts is 12 and at this deliverable we focused on most well-known companies. The remaining of the concepts will be handled in the following deliverable D2.3. Finally, Table 4 contains the selected concepts.

Concepts proposed explicitly by the user partners:			
• Electrolux logo • Bauknecht logo • BOSCH logo	• Philips logo • Siemens logo • Whirlpool logo	• General Electric logo • Hoover logo • Indesit logo	• LG logo • Miele logo • Samsung logo

Table 4: Selected concepts for the “Commercial media monitoring use case”.

7.5 Creation of training dataset

Based on the analysis of 7.4 section, the total number of concepts that we will deal with in this deliverable is 32. At this point, we should note that an essential step in the model development procedure is the creation of a training dataset. This training set must contain a significant number of images recognised as positive (relevant) in regard to a concept and number of images recognised as negative. The procedure we followed for creating the training dataset for each concept consists of the two following steps:

1. gathering of images related to each specific concept
2. manual annotation of the images returned by the APIs

The first step involves the downloading of image related to each concept using the Bing Image API⁴⁶, the flickr API⁴⁷ and Google Images⁴⁸. Appendix C1 contains a more extensive description on the image gathering step.

The second step involves improving the quality of the images downloaded (i.e. remove the wrongly returned images). Thus, a visual inspection and manual annotation of each image tag is realised. This annotation is realised through a java graphical interface that is depicted in Figure 9.

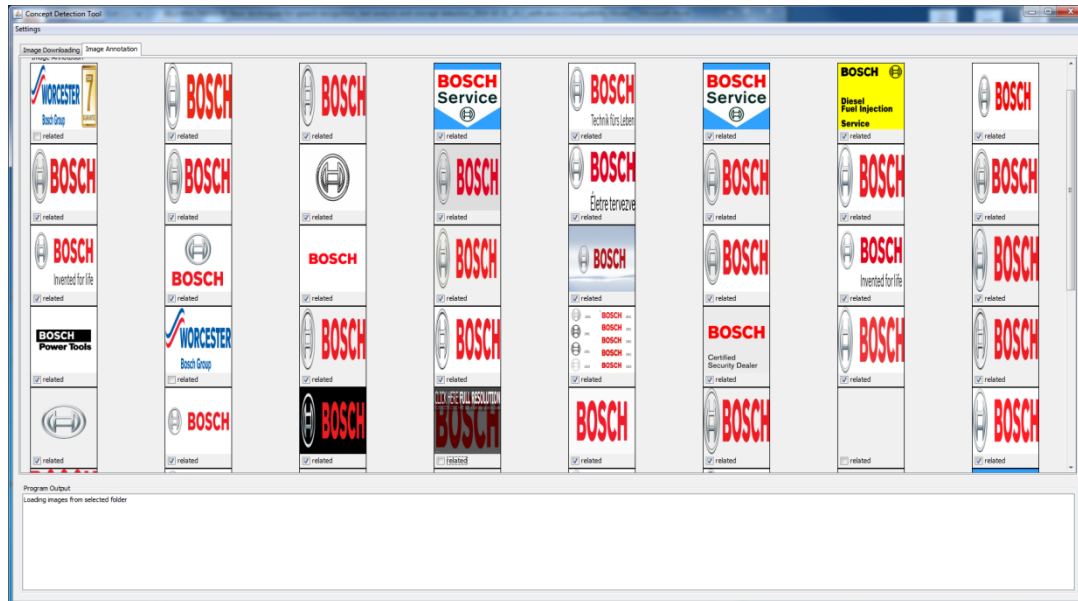


Figure 9: GUI for image manual annotation.

7.6 Evaluation

In this section, we will evaluate the multimedia concept detection module. It should be noted that the evaluation will be limited to:

1. The time performance of video decoding and the total procedure
2. The evaluation of the models developed for the concepts mentioned in section 7.4.

The evaluation in terms of accuracy of the steps constituting the module, including the shot segmentation, the scene segmentation, and the features and the classification in general will not be realised since the techniques used have been adequately tested and evaluated in several works (Sidiropoulos et al., 2011; Apostolidis et al., 2012; Stein et al., 2013; Apostolidis et al., 2014; Markatopoulou et al., 2014; Markatopoulou et al., 2015; Moumtzidou et al., 2010; Moumtzidou et al., 2011).

⁴⁶ <https://datamarket.azure.com/dataset/bing/search>

⁴⁷ <https://www.flickr.com/services/api/>

⁴⁸ <https://images.google.com/>

7.6.1 Time performance evaluation

In general, it is possible to evaluate the system in terms of time separately for each step and for complete module. Nonetheless in this deliverable we will focus only on comparing the time performance of the different techniques applied in the video decoding phase and on presenting some first results for the whole module as well. The comparison in terms of time for the feature extraction phase between local and global descriptors or between different descriptors in general will be realised in the next deliverable. The same applies for the classification phase, since at least for the basic version of the multimedia concept detection module we will use exclusively SVMs.

Dataset and infrastructure description

All the tests were realised on a PC with the following characteristics:

Processor	Intel Core i7 - 4770K CPU @ 3,50GHz
Memory	32 GB
System type	64-bit Operating System
Operation system	Windows 7 Professional - Service pack 1

Table 5: PC characteristics.

The dataset that was used includes a set of 8 video files, and each video differs either in terms of duration or dimensions (i.e. width and height of frames). The selection of the specific dataset was realised in order to allow us draw some conclusions on:

- how the video dimensions affect the computational time
- how the computational time increases with the increase of the video duration and,
- whether there is a limit in the video dimensions that can be handled beyond which the computational time is not acceptable

Table 6 contains the video files used during the testing procedure, their duration and dimension.

a/a	Video Name	Video Duration	Video Dimensions (width x height)
1	je20110329_atom10b_sd_avc.mp4	0:00:49	576 x 320
2	je20110329_atom10b_sd_sor.mp4	0:00:50	352 x 208
3	globalideas_en20120625_mongoleineu_sd_avc.mp4	0:06:25	576 x 320
4	gle20120813_patagonien_sd_sor.mp4	0:06:29	352 x 208
5	tt20131007_gesamt_sd_sor.mp4	0:26:03	512 x 288
6	2012_01_16_00_30_o_sd_sor.mp4	0:26:14	352 x 208
7	age20140204_gesamt_sd_avc.mp4	0:42:36	1280 x 720
8	age20140204_gesamt_sd_sor.mp4	0:42:36	512 x 288

Table 6: Video files.

Video decoding

As already mentioned, video decoding into keyframes can be realised by selecting frames with a predefined static time-step (i.e. fixed keyframe extraction) or by selecting the middle frame in a shot or even in a scene. Table 7 (and Figure C1 of the Appendix C2) contains the times needed in the three aforementioned methods for the dataset described earlier.

Video Ids	Time required (in seconds)			Number of keyframes	
	Fixed keyframe extraction	Shot segmentation	Scene segmentation	Fixed keyframe extraction	Shot segmentation
1	9	219.42	1	8	14
2	1	69.18	1	8	14
3	13	1768.1	2	64	82
4	9	512.04	2	64	65
5	48	4714.39	8	260	246
6	38	1666.56	8	262	276
7	146	49025.1	12	426	302
8	65	7632.03	12	426	290

Table 7: Time requirements and number of keyframes for different keyframe extraction methods.

After a careful observation of the Table 7, we can conclude that in case of fixed keyframe extraction, the time required is quite low. Things change radically when shot segmentation is applied. In this case, the experiments show that when the video dimensions are around 576x320 or less the time required for shot segmentation is twice or less the duration of the video, otherwise it can be very long especially when big videos are used. Thus a dimension of 576x320 or less is acceptable. Scene segmentation on the other hand, presupposes the application of shot segmentation and thus the time added after the application of shot segmentation is insignificant. However, it should be noted that new shot segmentation methods will be tested as well that might perform better in terms of time. Finally, as far as the number of keyframes extracted is concerned, when either fixed keyframe extraction or shot segmentation is applied, the number of keyframes extracted is comparable. However, the quality of the keyframes in terms of the video representation is not beyond any doubt different, since the first method is simply a random selection of keyframes while the second involves the use of image processing techniques.

Total Procedure

The time required to complete all the steps described depends also on the number of concepts/ models and of course the features extracted. At this point, we assume that in all cases all the aforementioned features (SIFT, opponent-SIFT, RGB-SIFT, SURF, opponent-SURF, RGB-SURF) will be extracted. Table 8 contains the time required for the total procedure to be realised in each one of the videos. A conclusion that can be easily drawn is that the number of concepts does not affect significantly the time required for the whole

procedure. Moreover, Table 8 (and Figure C2 of the Appendix C2) shows the time required for running the whole concept detection procedure in some videos with different duration.

Video Duration (in minutes)	Video dimensions	Time For Concept Detection for Different Number of Concepts (in minutes)		Time for Concept detection for Different Techniques of Keyframe extraction (in minutes)	
		23 Concepts	46 Concepts	Fixed segmentation	Shot segmentation
1	352 x 208	3,15	3,70	3,70	7,36
6	352 x 208	21,22	23,53	23,53	53
26	352 x 208	85,05	92,87	92,87	171,44
42	512 x 288	141,97	152,97	152,97	280,17

Table 8: Full time for fixed keyframe extraction and different number of concepts

Based on the values of Table 8, the number of videos that can be processed approximately per day using either of the two video decoding methods is shown in Table 9.

Video Duration (in minutes)	Number of videos processed per day	
	Fixed segmentation	Shot segmentation
1	360	180
6	57	26
26	15	8
42	9	5

Table 9: Number of videos processed from one PC in one day for video with different durations.

All the aforementioned results show that in general, the time required for video processing and concept detection is considerable. Thus, in order to be able to handle large amount of data, we plan on using Hadoop⁴⁹, which is an open-source software framework for storage and large-scale processing of data-sets on clusters of commodity hardware. We expect that with distributed computing technologies we will solve both the issue of time and capacity, since the concept detection requires also disk space for storing the intermediate results.

7.6.2 Model evaluation

In this section we present an evaluation of some of the concepts that were selected for each use case (see section 7.4). For each use case, we gathered a test dataset that was either retrieved from the internet (images retrieved from web pages) or provided by the user partners. These datasets were manually annotated and in the end the well-known IR metrics (precision, recall and f-score) were estimated for each concept. Since these metrics are adequately described in deliverable D8.1 (“Quality assurance and evaluation plan”), we will not provide any other details here.

⁴⁹ <http://hadoop.apache.org/>

Commercial media monitoring use case scenario

For this use case we have gathered 240 images in total that were classified in the concepts shown in Table 4. This dataset was manually annotated to 12 concepts and 20 relevant images were gathered for each concept. The results of the evaluation can be seen in Table 10. It is evident that the models developed can separate very well among the selected concepts. However, it should be noted that the images tested contained only logos and thus concept detection techniques that compare the global image characteristics were able to perform well. On the other hand, it should be noted, that the real data that will be used for MULTISENSOR do not depict only logos, quite the contrary the logos are only part of them. However, in such cases other techniques related to object detection should be studied that attempt to identify objects found inside images. Experiments in datasets with images containing logos as part of the image will be conducted in the advanced version of multimedia concept detection module which will be reported in the deliverable D2.3.

Concept name	Number of relevant images	Number of retrieved images	Number of retrieved relevant images	Precision	Recall	Fscore
Bauknecht logo	20	17	17	1	0.850	0.919
Bosch logo	20	18	18	1	0.900	0.947
Electrolux logo	20	19	19	1	0.950	0.974
General Electric logo	20	19	19	1	0.950	0.974
Hoover logo	20	19	19	1	0.950	0.974
Indesit logo	20	19	19	1	0.950	0.974
LG logo	20	19	19	1	0.950	0.974
Miele logo	20	20	20	1	1.000	1.000
Philips logo	20	19	19	1	0.950	0.974
Samsung logo	20	18	18	1	0.900	0.947
Siemens logo	20	20	20	1	1	1
Whirlpool logo	20	17	15	0.882	0.750	0.811

Table 10: Evaluation metrics values for the “Commercial media monitoring use case” dataset.

Journalism use case scenario

For this use case, we present the evaluation results for the concepts shown in Table 2. Specifically, as far as the concepts that are defined from TRECVID, we present the results of ITI-CERTH’s team participating in TRECVID 2014 SIN task that were published in (Markatopoulou et al., 2013). The results of the experiments can be found in Table 11 and they are shown in terms of Extended Inferred Average Precision (MXinfAP) (Yilmaz et al., 2008), which is an approximation of the Average Precision (AP) suitable for the partial ground truth that accompanies the TRECVID dataset (Smeaton et al., 2006), and precision at the first 10 and 100 results. Regarding the Average Precision, it is a metric that considers the

ranking for the returned results as well. Thus, for systems that return a ranked sequence of results, it is desirable to also consider the order in which the returned documents are presented. By computing a precision and recall at every position in the ranked sequence of documents, one can plot a precision-recall curve, plotting precision $p(r)$ as a function of recall r . The AP computes the average value of $p(r)$ over the interval from $r=0$ to $r=1$.

It should be noted that TRECVID does not provide participants with evaluation information for all 500, since it evaluates only 60 concepts each year, and thus Table 11 contains the only part of the concepts selected for MULTISENSOR. Specifically, there is no available evaluation information for the concepts: bicycle, building, car, cityscape, factory, factory worker, laboratory, landscape, scientists and waterscape. Regarding the performance of the concepts, it should be noted that ITI-CERTH team had achieved results (the ones shown in Table 11) that ranked well above the average compared to the other participating teams.

Concept Name	Extended Inferred Average Precision	Precision @ 10 res	Precision@100
Airplane	0.092	0.3	0.217
Anchorperson	0.574	0.6	0.777
Boat Ship	0.261	0.9	0.81
Demonstration/ Protest	0.194	0.6	0.457
Explosion/ Fire	0.119	0.8	0.64
Forest	0.091	0.3	0.3
Government Leader	0.367	1.0	0.8
Press Conference	0.233	0.9	0.803
Reporters	0.0119	0.0	0.06

Table 11: Evaluation metrics values for the “Journalism use case” dataset and TRECVID concepts from ITI-CERTH participation in TRECVID 2014.

Finally, we have gathered 200 images in total that were classified in the concepts shown in Table 2 that are not part of the TRECVID concepts. This dataset was manually annotated to 10 concepts, and 20 relevant images were gathered for each concept. The results of the evaluation can be seen in Table 12 and it is evident that the models developed can separate very well among the selected concepts. However, it should be noted that the real images retrieved from web pages or videos or any other sources used by MULTISENSOR will be far more complicated. Thus, in regards to the concepts related to logos (e.g. EnBW logo, E-on logo), and similarly to the previous use case, we will apply object detection techniques that attempt to identify objects found inside images, while as far as the other concepts are concerned (e.g. Lattice tower), more realistic datasets will be used for evaluation purposes. These extended experiments along with the new techniques studied will be realised in the next deliverable D2.3

Concept name	Number of relevant images	Number of retrieved images	Number of retrieved relevant images	Precision	Recall	Fscore
EnBW logo	20	14	14	1	0.7	0.824

E-On logo	20	14	14	1	0.7	0.824
Nuclear energy logo	20	15	15	1	0.75	0.857
RWE logo	20	19	18	0.947	0.9	0.923
Vattenfall logo	20	12	12	1	0.6	0.75
Lattice tower	20	19	19	1	0.95	0.974
Recycle bin	20	20	20	1	1	1
Smog	20	20	20	1	1	1
Solar panel	20	20	20	1	1	1
Wind turbine	20	19	19	1	0.95	0.974

Table 12: Evaluation metrics values for the “Journalism use case” dataset

8 MACHINE TRANSLATION

8.1 Management summary

Automatic machine translation (MT) is employed within the MULTISENSOR project with two main goals:

- To provide the translation of the summarisation results in the end of the content analysis and summarisation chain.
- To enable full-text translation on-demand during the development of language dependent analysis tools in the project, in case a subset of required languages is not supported by these tools (i.e. as a workaround, until all required languages are supported by the respective tools).

In the first case, the translations will be produced at the end of the analysis/summarisation process and will be stored together with the summaries. In the second case, the translations produced by MT provide the input for the text analysis chain and follow the same analysis procedure as the input from original text sources in the required language.

The languages covered by MT in the MULTISENSOR project will be English, German, Spanish, Bulgarian, and French.

8.2 State of the art

In the beginning, the machine translation systems followed a simple substitution of words from one language by the corresponding words in another language. But this approach has not lead to good translation results. It became clear that the translation requires not only a lexicon (words and their translations) but rather a profound knowledge of specific structures in the source language and a way to generate corresponding structures in the target language. Two main approaches to achieve this have been developed:

- Knowledge-driven: based on exploiting the expert experience of lexicographers and linguists, who write lexicons and grammar rules manually, founded on their own knowledge (RBMT = rule-based machine translation).
- Data-driven: based on automatic corpus analysis and statistical learning methods, where both translations and the rules are learned from a sufficient amount of bilingual corpora, i.e. from existing translations (SMT = statistical MT).

Today, the field of machine translation is characterised by a strong dominance of the statistical techniques. The most important prerequisite for the development of probabilistic-based systems is the availability of parallel corpora, i.e. of texts in two or more languages that are perfect (human) translations of each other. Many open SMT tools exist that offer software for training, tuning and decoding tasks. One of the widely used ones is Moses⁵⁰, a toolkit originally developed by Philipp Koehn at the University of Edinburgh, and continuously improved and further developed by a growing number of researchers all over the world.

⁵⁰ <http://www.statmt.org/moses/>

8.3 Technical framework and approach

For MULTISENSOR, we adopt the SMT approach and the machine learning techniques associated to it. Since the Moses framework is a language-independent platform, we use it for the development of all project MT pairs in the same way.

Moses offers:

- A training component for translation models (bilingual phrase tables) and for reordering models (bilingual “rules” how to deal with different word orders in source and target language).
- A tuning component for better adjusting the translation parameters according to the given domain or text genre; it is based on an extra set of human translations that is representative for the intended translation task in terms of vocabulary and style.
- A decoding component with powerful search algorithms that computes all possible translation hypotheses and finds the translation with the highest probability.
- Support for externally estimated LMs and integration of LM tools (SRILM, IRSTLM etc.), as well as already integrated LM software (KenLM, RandLM).
- A work-flow control tool (EMS = experiment management system), which enables the user to automatise the whole training-tuning-evaluation chain; it offers a graphical interface to follow the progress of each step visually.

Moses, like many other SMT platforms, supports different kinds of translation models:

- **Phrase-based:** this is the “standard” and most widely used approach. Instead of learning the translation word by word, larger word sequences (currently, up to 7 words) are being taken into account; thus, larger contexts, different word orders in source and target, as well as distant dependencies are considered. However, very long dependences remain unseen and cannot be learned; this is one of the major problems and shortcomings in the phrase-based approach. Moses also implements an extension to the phrase-based approach, known as factored translation. It enables extra linguistic information (such as part of speech, morphological information, semantics etc.) to be added to phrase-based models, without applying deeper linguistic syntax rules.
- **Hierarchical:** while the translation units (phrases) in the phrase-based models are random and not linguistically motivated at all, the phrases learned in the hierarchical approach are more “syntactically-inspired”. Group of words (chunks) are replaced by higher (non-terminal) terms. They can reflect nominal, adjectival or any other syntactically motivated groups of words, but not higher relations between them, i.e. this approach does not use linguistic syntax rules.
- **Tree-based:** while the phrase-based and hierarchical models map source phrases onto the target phrases, the tree-based (syntactic) models operate on rules, which are based on syntactic structures of language.

In MULTISENSOR we have adopted the phrase-based model.

8.4 Implementation and components

The language pairs covered in the project will be German, French, Spanish, and Bulgarian into and from English (eight direct translation directions). When using English as a pivot

language, 12 additional translation directions (e.g. German-French, Bulgarian-Spanish) can be supported:

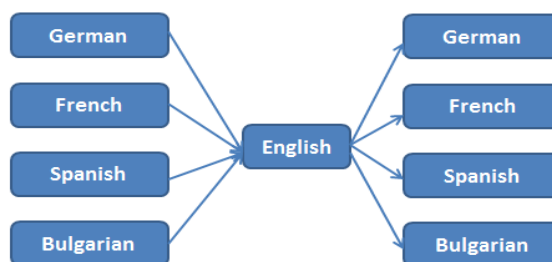


Figure 5: MT translation directions in MULTISENSOR

In the first year of the project, baseline systems for all languages into English have been developed. The other directions (from English into the other languages) as well as advanced versions of all systems will be released during the second year.

8.4.1 Resources

While the software components are basically language-independent, the language-specific system development requires many different bilingual and monolingual resources:

- Bilingual: parallel training corpora, parallel reference translations for tuning and testing purposes, bilingual lexicons.
- Monolingual: huge amounts of data in the target language for language modelling, many linguistic resources and lexica for normalisation and pre-processing task.

The baseline systems have been trained on freely available open source corpora, such as Europarl, DGT, JRC, and SETimes⁵¹ for Bulgarian (cf. Tyers, 2010) and on parallel corpora crawled by Linguatrec. All of them can be consider general-domain. The advanced versions will be domain-adapted for the MULTISENSOR use cases and domains, based on as much in-domain parallel data as the use case partners can provide.

8.4.2 Data preparation

The data preparation process is very similar to the corpus collection and pre-processing steps described above in the ASR section.

One part of the data was already in form of aligned corpora, i.e. bilingual translation memories (Europarl, DGT) and as such it did not undergo the whole pre-processing chain. The other part, e.g. crawled data has been prepared as described in the following.

Crawler and HTML Parser

In comparison to the monolingual crawler described in section 6.4.1, the bilingual crawler offers the following additional features:

- When crawling web sites, it first looks for pages with an identifiable language flag
- For each found page it searches a corresponding parallel page (in another language)

⁵¹ <http://www.setimes.com>

- In case such pages have been found, the crawler performs some checks: Does the translation contain a minimum of text? Do the pages indeed contain different and required languages? Are the texts most probably translations of each other (similar length, same numbers, same named entities, etc.)?

The crawled documents are stored in HTML-format, and then offline processed by the HTML-parser. Additionally to the functionality described in 6.4.1, the parser, when processing parallel documents, tries to align them at least on the paragraph level. This enables a better alignment on the sentence level in the next steps.

Sentence splitter and tokeniser

These components have the same functionality as described in section 6.4.1. However, in order to really be able to find parallel sentences, it is very important to harmonise the segmentation in both languages and to use similar heuristics for the detection of sentence boundaries (e.g. semicolon as sentence end in both languages). Tokeniser detaches punctuation marks, symbols, brackets etc. from words, but in contrast to the ASR, none of them are removed from the training corpus. Both tools have been developed by Linguatrec.

Sentence aligner

Alignment aims at identifying corresponding sentences in two given parallel texts. There exist many open-source or commercial alignment tools. Some of them take original texts as input and have integrated sentence segmentation and tokenisation tasks. However, our experiments have shown better results if the input to the alignment task is already pre-processed and segmented as described above. Best results we achieved with the Hunalign⁵², an open-source language-independent alignment tool, which also can make use of bilingual dictionaries if any are provided. In absence of bilingual dictionaries, Hunalign builds an automatic dictionary during a first alignment pass, and realigns the sentences in a second pass, using this dictionary.

Final cleaning and casing

In a last step before training we filter out sentences: (a) with a length of more than 60 words (b) if the sentence length ratio between source and target is higher than 3x (c) if sentences belong to a third language (d) if sentence pairs are very different in terms of occurrence of numbers, special characters and symbols.

In the current baseline version, all training data have been used in their original casing. Only the sentence beginnings have been true-cased, i.e. proper names in all languages and also nouns in German have been kept capitalised; all other words have been converted to lower case. Should tests show that too many errors regarding the capitalisation occur in the translation, we will consider another approach: to lower-case the training data, and to use a re-casing component (rule-based or trained statistically) to restore the correct orthography.

Language modelling

For language modelling we use the same tools and procedures as described in 6.4.2 above to produce n-gram LMs. The advantage of n-gram LMs is that such models are corpus-based

⁵² <http://mokk.bme.hu/resources/hunalign/>

and as such reflect the “real world”. They are in particular suitable for languages with a stronger word order, since they perfectly reproduce local syntax and also semantics (usually, LMs with up to 3 or 4-grams are used in MT). The disadvantage of 3/4-gram LMs is that long-distance dependencies cannot be reflected and the languages with flexible word order are more difficult to be mapped into a model. However, there is no much sense in using longer order, since it would make the language models very huge and slow, while the probability for longer and longer sequences (6, 7 or more) to be seen again is getting lower and lower.

8.5 Evaluation

For machine translation quality there are three main assessment methodologies:

- **Automatic metrics:** There exist many tools for automatic assessment of the translation quality, such as BLEU (Papineni et al. 2002), NIST⁵³, or METEOR (Denkowski, Lavie 2011). They measure how similar is the MT output to one or several reference translations. For this, test sets with “ideal” translations are prepared which are as domain-specific as possible and representative for the intended translation tasks.
- **Manual comparative evaluation:** human evaluators manually compare two MT outputs (translations of the same sentence) and decide which one is better.
- **Manual absolute evaluation:** evaluators compare the translation with its source sentence and give assessment regarding adequacy, fluency and intelligibility of the translation.

In MULTISENSOR, we will use all three evaluation approaches. The automatic metrics are in particular suitable for a fast and continuous monitoring of changes between two versions. The manual evaluation will be done in the end of the project; being a very time-consuming and cost-intensive approach, it cannot be applied after each new development cycle. We will use two different kinds of test sets: (a) in-domain data, ca. 2000 sentences for each language pair. We consider it affordable to create test sets for the UC1-scenario 1 (press articles about energy policies). It will be more difficult to collect reference translation for the other use cases and scenarios (house appliances, and yoghurt export). Should the parallel data provided by the user partners not cover all use cases, we will use the same test set for all scenarios (b) general test sets, originating from another EU project (e.g. EUROMATRIX⁵⁴), in order to compare our results with the results achieved in similar projects.

8.6 Integration into the MULTISENSOR platform

The machine translation system runs on Linguatrec servers. The communication between the MT server and MULTISENSOR platform will be established via REST web services.

⁵³ <ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a-20091001.tar.gz>

⁵⁴ <http://www.euromatrix.net/>

9 CONCLUSIONS

This deliverable reports on the basic techniques for named entities recognition, concept extraction from text, concept linking and relations, speech recognition, multimedia concept detection, and machine translation.

The **named entities recognition** module identifies names of persons, locations, companies and institutions, as well as amounts and dates. Whenever possible, additional attributes such as first name/ surname for persons, type of location, prices and subclass of amounts etc. are recognised. Our NER approach is knowledge-driven, based on language-dependent lexicons and grammars. The first baseline version of NER module for English, based on lexicon lookup, has been released; lexicons for German, French, and Spanish have been produced, and the collection of Bulgarian resources is ongoing. The next version will include grammars, which will allow recognition of entities not present in the lexicons. Furthermore, the integration of the NIF ontology and disambiguation of named entities against DBPedia is ongoing. The NER component runs on Linguattec server. The service call includes, beside the NE recognition, also the sentence splitting and tokenisation modules.

Concerning the **concept extraction** and **concept linking and relation extraction modules**, an annotation exercise is being carried out with the goal to assess the coverage of existing datasets and lexical resources and establish criteria for what contents should be extracted. The annotation will result in a set of annotation guidelines, a draft of which is presented in A.3, and a manually annotated gold standard to be used in the evaluation of the models. The general approach to relation extraction is based mostly on a multilingual deep dependency parser capable of identifying predicates and their arguments in the text. As described in D6.1, the training resources needed to deploy parsing models for the English and Spanish languages are ready, while corpora for the remaining languages will be released in the next few months.

The **speech recognition** module is available for German and English. It is based on open-source framework RWTH-ASR. In the first year of the project, baseline systems have been produced based on online freely available contents; they ensure wide general-domain recognition. In the second year of the project, domain-adapted systems will be trained, based on project-relevant in-domain data. The first versions of German and English systems differ strongly in their volume and size: while the English recognition lexicon contains about 150.000 entries, has the German lexicon almost 600.000 entries. The size of the language models in binary format is ca. 500 MG for English and ca. 6 GB for German. Doing so, we intend to answer the question of how the size of the recognition data correlates to the recognition quality. The ASR module runs on Linguattec server.

As far as **multimedia concept extraction** is concerned, this deliverable presents the first and basic version of the related techniques. In this context, we have created a framework that implements the concept testing procedure and incorporates a video decoding step, a feature extraction step and a classification step. Regarding the video decoding and feature extraction steps, we have used the most recent approaches in the relevant literature while for the classification step we tested the well-established SVM classifiers. In order to evaluate the process proposed, we run evaluation experiments that test both the time and the accuracy of the retrieved results. The most important conclusion that was drawn from the time evaluation was that significant time and disk storage is required and thus it is essential

that we move to distributed computing technologies for handling real data. Moreover, as far as the evaluation of the retrieved results is concerned, we have conducted several tests that performed satisfactory but the testing dataset that was used was not very realistic since it did not contain very complicated images. In order to be able to handle such images, it is expected to test object detection techniques as well that are able to locate specific objects within an image. Finally, in order to train the models used for capturing concepts related to MULTISENSOR, we developed an annotation tool that required manual annotation of the images. However, in order to limit the time required for completing this task (approximately 2 hours are required for annotating 2000 images), we plan on applying image processing techniques such as clustering at a later stage that will limit this time and therefore allow the annotation of bigger datasets and of more concepts. The aforementioned improvements will be presented in the two next deliverables D2.3 and D2.4.

Machine translation in MULTISENSOR project is statistical based and it uses the open-source framework Moses. We have adopted the phrase-based approach. Eight direct translation directions will be developed (German, French, Spanish and Bulgarian from and into English). When using English as a pivot language, twelve additional non-direct language directions will be supported. In the first year of the project the baseline systems for the translation into English have been released. The training data consisted of freely available bilingual corpora and crawled web data; they should support a rather general-domain translation. Depending on the availability of parallel project-relevant data, the next versions will be adapted for the project use cases. The MT module runs on Linguatrec server.

10 REFERENCES

- Albatal, R., and Mulhem, P. 2010. "MRIM-LIG at ImageCLEF 2010 Visual Concept Detection and Annotation task", In CLEF.
- Apostolidis, E., Dimopoulos, M., Mezaris, V., Stein, D., Blom, J., Lašek, I., Sahuguet, M., Huet, B., de Abreu Pereira, N., Müller, J. 2012. "D1.1 - State of the Art and Requirements Analysis for Hypervideo", Public Deliverable, The LinkedTV Project (FP7-ICT-2011-7), <http://www.slideshare.net/linkedtv/linked-tv-d11state-of-the-art-and-requirements-analysis-for-hypervideo>.
- Apostolidis, E., Mezaris, V., Sahuguet, M., Huet, B., Cervenková, B., Stein, D. 2014. "Automatic Fine-grained Hyperlinking of Videos within a Closed Collection using Scene Segmentation", 22nd ACM International Conference on Multimedia, Orlando, Florida, USA.
- Ariki, Y., Kumano, M., and Tsukada, K. 2003. "Highlight scene extraction in real time from baseball live video", In Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval, MIR '03, pp. 209–214, New York, NY, USA. ACM.
- Augenstein, I., & Rudolph, S. 2012. "LODifier: Generating Linked Data from Unstructured Text", In Proceedings of the 9th international conference on The Semantic Web: research and applications (ESWC'12), pp. 210-224, Springer-Verlag Berlin, Heidelberg.
- Azzam, S., Humphreys, K., Gaizauskas, R. 1998. "Evaluating a Focus-based Approach to Anaphora Resolution", Proc. Joint COLING-ACL.
- Baber, J., Afzulpurkar, N., and Bakhtyar, M. 2011. "Video segmentation into scenes using entropy and surf", 7th International Conference on Emerging Technologies (ICET), pp. 1–6.
- Ballesteros, M., and Bohnet, B. 2014. "Automatic Feature Selection for Agenda-Based Dependency Parsing", 25th International Conference on Computational Linguistics (COLING 2014) Dublin, Ireland.
- Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. 2008. "Speeded-Up Robust Features (SURF)", Comput. Vis. Image Underst., vol. 110(3), pp. 346–359.
- Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. 2008. "Speeded-up robust features (surf)", Comput. Vis. Image Underst., vol. 110(3), pp. 346–359.
- Bel, N., Poch, M., Toral, A. 2012. "PANACEA (Platform for Automatic, Normalised Annotation and Cost-Effective Acquisition of Language Resources for Human Language Technologies)", Proceedings of the 16th Annual Conference of the European Association for Machine Translation, Trento, Italy.
- Bisani, M., Ney, H. 2005. "Open Vocabulary Speech Recognition with Flat Hybrid Models", In Interspeech, pp. 725–728.
- Bisani, M., Ney, H. 2008. "Joint-sequence models for grapheme-to-phoneme conversion", Speech Communication, vol. 50, No. 5, pp. 434–451.
- Bohnet, B., and Nivre, J. 2012. "A Transition-Based System for Joint Part-of-Speech Tagging and Labeled Non-Projective Dependency Parsing", EMNLP-CoNLL.

- Bos, J. 2008. "Wide-Coverage Semantic Analysis with Boxer". In: Bos, J., Delmonte, R. (eds.) *Semantics in Text Processing. STEP 2008 Conference Proceedings. Research in Computational Semantics*, vol. 1, pp. 277–286. College Publications.
- Bosch, A., Zisserman, A., and Munoz, X. 2008. "Scene classification using a hybrid generative/discriminative approach", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30(4), pp. 712–727.
- Bouayad-Agha, N., Casamayor, G., Wanner, L., and Mellish, C. 2013. "Overview of the First Content Selection Challenge from Open Semantic Web Data", In: *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 98–102, Sofia, Bulgaria.
- Bradski, G. 2000. "The OpenCV Library", *Dr. Dobb's Journal of Software Tools*.
- Burghouts, G. J., and Geusebroek, J.-M. 2009. "Performance evaluation of local colour invariants", *Comput. Vis. Image Underst.*, vol. 113(1), pp. 48–62.
- Carreras, X., Màrquez, L., Padró, L. 2003. "Named Entity Recognition for Catalan Using Spanish Resources", *Proc EACL*.
- Chang, S.-F., Ellis, D., Jiang, W., Lee, K., Yanagawa, A., Loui, A. C., and Luo, J. 2007. "Large-scale multimodal semantic concept detection for consumer video", In *Proceedings of the 2007 International Workshop on Multimedia Information Retrieval, MIR '07*, pp. 255–264, New York, NY, USA. ACM.
- Chang, S.-F., Sikora, T., and Purl, A. 2001. "Overview of the MPEG-7 standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11(6), pp. 688–695.
- Charniak, E. (1997): *Statistical Parsing with a Context-free Grammar and Word Statistics*. Proc AAAI.
- Chasanis, V., Kalogeratos, A., and Likas, A. 2009. "Movie segmentation into scenes and chapters using locally weighted bag of visual words", In *Proceedings of the 2009 ACM International Conference on Image and Video Retrieval, CIVR '09*, pp. 35:1–35:7, New York, NY, USA. ACM.
- Chatfield, K., Lempitsky, V., Vedaldi, A., Zisserman, A. 2011. "The devil is in the details: an evaluation of recent feature encoding methods", In: *British Machine Vision Conference*. pp. 76.1-76.12. British Machine Vision Association.
- Chen, L., Rizvi, S. J., Tamer zsu, M., and Tamer, M. 2003. "Incorporating audio cues into dialog and action scene extraction", In *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases*, pp. 252–264.
- Cunningham, H., Maynard, D., Bontcheva K., Tablan, V. 2002. "GATE: A framework and graphical development environment for robust NLP tools and applications", In *Proc. 40th ACL*.
- Cunningham, H., Maynard, D., Bontcheva, K. 2010. "Developing Language Processing Components with GATE Version 5 (a User Guide)", The University of Sheffield, 01.
- Curran, J.R., Clark, S., Bos, J. 2007. "Linguistically Motivated Large-Scale NLP with C&C and Boxer", In: *Proceedings of the ACL 2007 Demo Session*, pp. 33–36.

- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. 1990. "Indexing by latent semantic analysis", *Journal of the American Society for Information Science*, vol. 41(6), pp. 391–407.
- Delezoide, B., Precioso, F., Gosselin, P., Redi, M., Merialdo, B., Granjon, L., Pellerin, D., Rombaut, M., Jegou, H., et al. 2011. "Irim at trecvid 2011: Semantic indexing and instance search", In *Proceedings of the 9th TRECVID Workshop*, Gaithersburg, USA.
- Denkowski, M., Lavie A. 2011. "Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems", In: *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, Edinburgh.
- Doulaverakis, C., Vagionitis, S., Zervakis, M., and Petrakis, E. 2004. "Adaptive methods for motion characterization and segmentation of MPEG compressed frame sequences", In: Aurlio Campilho and Mohamed Kamel (eds.), *Image Analysis and Recognition*, vol. 3211, pp. 310–317.
- Draicchio, F., Gangemi, A., Presutti, V., & Nuzzolese, A. G. 2013. "FRED: From Natural Language Text to RDF and OWL in One Click", In: *The Semantic Web: ESWC 2013 Satellite Events*, *Lecture Notes in Computer Science*, Volume 7955, 2013, pp 263-267.
- Exner, P., & Nugues, P. 2012. "Entity extraction: From unstructured text to DBpedia RDF triples", In *The Web of Linked Entities Workshop (WoLE 2012)*.
- Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. 2008. "LIBLINEAR: A Library for Large Linear Classification", *Journal of Machine Learning Research*, vol. 9, pp. 1871-1874.
- Federico, M., Bertoldi, N., Cettolo, M. 2008. "IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models", *Proceedings of Interspeech*, Brisbane, Australia
- Forssn, P.-E, (2007): "Maximally stable colour regions for recognition and matching", In *CVPR '07*. IEEE Computer Society.
- Gaizauskas, R., Humphreys, K. 1997. "Quantitative Evaluation of Coreference Algorithms in an Information Extraction System", *Technical Report Univ. Sheffield*.
- Gales, M, Young, S. 2007. "The Application of Hidden Markov Models in Speech Recognition", *Foundations and Trends in Signal Processing*, vol. 1, no 3, pp. 195-304.
- Gao, X., Xiao, B., Tao, D., and Li, X. 2008. "Image categorization: Graph edit distance+edge direction histogram", *Pattern Recognition*, vol. 41(10), pp. 3179–3191.
- Hahn, S. and Rybach, D. 2011. "Building an Open Vocabulary ASR System using Open Source Software", *Interspeech*, Florence, Italy.
- Harris, C., and Stephens, M. (1988): "A combined corner and edge detector", In *Proc. of 4th Alvey Vision Conference*, pp. 147–151.
- Hauptmann, A. G., Chen, M.-Y., Christel, M., Das, D., Lin, W.-H., Yan, R., Yang, J., Backfried, G., and Wu, X. 2006. "Multi-lingual broadcast news retrieval", In *Proc. of TRECVID*.
- Heafield, K. 2011. "KenLM: Faster and smaller language model queries", In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, Edinburgh.

- Hradis, M., Reznicek, I., Behun, K., and Otrusina, L. 2011. "Brno university of technology at trecvid 2011 sin, ccd", In Proceedings of the 9th TRECVID Workshop, Gaithersburg, USA.
- Hua, X.-S., Lu, L., and Zhang, H.-J. 2004. "Optimization-based automated home video editing system", IEEE Transactions on Circuits and Systems for Video Technology, vol. 14(5), pp. 572–583.
- Huiskes, M. J., Thomee, B., and Lew, M. S. 2010. "New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative", In Proceedings of the 2010 International Conference on Multimedia Information Retrieval, MIR '10, pp. 527–536, New York, NY, USA. ACM.
- Iurgel, U., Meermeier, R., Eickeler, S., and Rigoll, G. 2001. "New approaches to audio-visual segmentation of TV news for automatic topic retrieval", In Proceedings of the 2001 IEEE International Conference on the Acoustics, Speech, and Signal Processing, vol. 3, pp. 1397–1400, Washington, DC, USA. IEEE Computer Society.
- Jegou, H., Douze, M., Schmid, C., Perez, P. 2010. "Aggregating local descriptors into a compact image representation", IEEE on Computer Vision and Pattern Recognition (CVPR 2010). pp. 3304–3311. San Francisco, CA.
- Jegou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., Schmid, C. 2012. "Aggregating local image descriptors into compact codes", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34(9), pp. 1704–1716.
- Jiang, J., Rui, X., and Yu, N. 2008. "Feature annotation for visual concept detection in Image-CLEF 2008", In Workshop: Working Notes for the CLEF 2008 Workshop.
- Jiang, Y.-G., Ngo, C.-W., and Yang, J. 2007. "Towards optimal bag-of-features for object categorization and semantic video retrieval", In Proceedings of the 6th ACM international conference on Image and video retrieval, CIVR '07, pp. 494–501, New York, NY, USA. ACM.
- Jiang, Y.-G., Yang, J., Ngo, C.-W., and Hauptmann, A.G. 2010. "Representations of keypointbased semantic concept detection: A comprehensive study", IEEE Transactions on Multimedia, vol. 12(1), pp. 42–53.
- Jiang, Y.-G., Zhao, W.-L., and Ngo, C.-W. 2006. "Exploring semantic concept using local invariant features", In Proceedings of Asia-Pacific Workshop on Visual Information Processing.
- Jurie, F., and Triggs, B. 2005. "Creating efficient codebooks for visual recognition", In 10th IEEE International Conference on Computer Vision, ICCV '05, vol. 1, pp. 604–610.
- Kaumanns, D. 2011. "Strategies of a Biparallel Crawler System", Internal document, Linguattec GmbH.
- Koehn, P. 2005. "Europarl: A Parallel Corpus for Statistical Machine Translation", MT Summit 2005.
- Koehn, P., Hieu H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. 2007. "Moses: Open source toolkit for statistical machine translation", In ACL, demonstration session.

- Koehn, P., Schroeder, J. 2007a. "Experiments in Domain Adaptation for Statistical Machine Translation", In Proceedings of the Second Workshop on Statistical Machine Translation. Prague, pp. 224-227.
- Koehn, P. 2010. "Statistical Machine Translation", Cambridge University Press.
- Lawrie, D., Mayfield, J., McNamee, P., Oard, D.W. 2012. "Creating and Curating a Cross-Language Person-Entity Linking Collection", Proc. LREC Istanbul.
- Le, D.-D., and Satoh, S. 2009. "Efficient concept detection by fusing simple visual features", In Proceedings of the 2009 ACM symposium on Applied Computing, SAC '09, pp. 1839–1840, New York, NY, USA. ACM.
- Li, F.-F., and Perona, P. 2005. "A bayesian hierarchical model for learning natural scene categories", In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR '05, vol. 2, pp. 524–531, Washington, DC, USA.
- Liao, J., and Zhang, B. 2007. "A robust clustering algorithm for video shots using Haar wavelet transformation", In Proceedings of SIGMOD2007 Ph.D. Workshop on Innovative Database Research (IDAR2007), Beijing, China.
- Ling, X., Yuanxin, O., Huan, L., and Zhang, X. 2008. "A method for fast shot boundary detection based on SVM". Congress on Image and Signal Processing (CISP '08), vol. 2, pp. 445–449.
- Liu, S., Zhu, M., and Zheng, Q. 2008. "Video shot boundary detection with local feature post refinement", 9th International Conference on Signal Processing (ICSP 2008), pp. 1548–1551.
- Lowe, D. G. 2004. "Distinctive image features from scale-invariant keypoints", Int. J. Comput. Vision, vol. 60(2), pp. 91–110.
- Lowe, D. G. 2004. "Distinctive image features from scale-invariant keypoints", International Journal of Computer Vision, vol. 60, pp. 91–110.
- Lu, H., Li, Z., and Tan, Y.-P. 2004. "Model-based video scene clustering with noise analysis", In Proceedings of the 2004 International Symposium on Circuits and Systems, ISCAS '04, vol. 2, pp. 105–108.
- Maji, S., Berg, A.C., and Malik, J. 2008. "Classification using intersection kernel support vector machines is efficient", In Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.
- Markatopoulou, F., Mezaris, V., Kompatsiaris, I. 2014. "A comparative study on the use of multi-label classification techniques for concept-based video indexing and annotation", In: Gurrin, C., Hopfgartner, F., Hurst, W., Johansen, H., Lee, H., OConnor, N. (eds.) MultiMedia Modeling. LNCS, vol. 8325, pp. 1-12. Springer.
- Markatopoulou, F., Moumtzidou, A., Tzelepis, C., Avgerinakis, K., Gkalelis, N., Vrochidis, S., Mezaris, V., and Kompatsiaris, I. 2013. "ITI-CERTH participation to TRECVID 2013," in TRECVID 2013 Workshop, Gaithersburg, MD, USA, 2013.
- Markatopoulou, F., Pittaras, N., Papadopoulou, O., Mezaris, V., Patras, I. 2015. "A Study on the Use of a Binary Local Descriptor and Color Extensions of Local Descriptors for Video

Concept Detection", Proc. 21th Int. Conf. on MultiMedia Modeling (MMM'15), Sidney, Australia.

Marszalek, M., Schmid, C., Harzallah, H., and van de Weijer, J. 2007. "Learning object representations for visual object class recognition", Visual Recognition Challenge workshop, in conjunction with ICCV.

Matas, J., Chum, O., Urban, M., and Pajdla, T. 2002. "Robust wide baseline stereo from maximally stable extremal regions", In British Machine Vision Conference, vol. 1, pp. 384–393.

McCamy, C., Marcus, H., and Davidson, J. 1976. "A color-rendition chart", Journal of Applied Photographic Engineering, vol. 2(3), pp. 95–99.

McCulloch, D., Campbell, C. "An Investigation into Novelty Detection", http://www.enm.bris.ac.uk/teaching/projects/2004_05/dm1654/svm_classification.html

Mezaris, V., Dimou, A., and Kompatsiaris, I. 2010. "On the use of feature tracks for dynamic concept detection in video", In 17th IEEE International Conference on Image Processing, ICIP '10, pp. 4697–4700.

Mezaris, V., Kompatsiaris, I., and Strintzis, M. G. 2004. "Still image segmentation tools for objectbased multimedia applications", International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, pp. 701–725.

Mikolajczyk, K., and Matas, J. 2007. "Improving descriptors for fast tree matching by optimal linear projection", In IEEE 11th International Conference on Computer Vision, ICCV '07, pp. 1–8.

Mille, S., Burga, A., and Wanner, L., 2013. "Ancora-UPF: A Multi-Level Annotation of Spanish", The 2nd International conference on Dependency Linguistics (DEPLING 2013).

Moosmann, F., Triggs, B., and Jurie, F. 2006. "Fast discriminative visual codebooks using randomized clustering forests", In Bernhard Scholkopf, John Platt, and Thomas Hoffman, editors, NIPS, pp. 985–992. MIT Press.

Moumtzidou, A., Dimou, A., Gkalelis, N., Vrochidis, S., Mezaris, V., and Kompatsiaris, I. 2010. "ITI-CERTH participation to TRECVID 2010", In TRECVID 2010 Workshop.

Moumtzidou, A., Sidiropoulos, P., Vrochidis, S., Gkalelis, N., Nikolopoulos, S., Mezaris, V., Kompatsiaris, I., and Patras, I. 2011. "ITI-CERTH participation to TRECVID 2011", In TRECVID 2011 Workshop, Gaithersburg, MD, USA, 12/2011.

Mylonas, P., Spyrou, E., Avrithis, Y., and Kollias, S. 2009. "Using visual context and region semantics for high-level concept detection", Trans. Multi., vol. 11(2), pp. 229–243.

Nadeau, D., Sekine, S. 2007. "A survey of named entity recognition and classification", Lingvisticae Investigationes, vol. 30(1), pp. 3–26.

Ney, H., Welling, L., Ortmanns, S., Beulen, K., and Wessel, F. 1998. "The RWTH Large Vocabulary Continuous Speech Recognition System", In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pages 853–856, Seattle, WA, USA.

Niaz, U., Redi, M., Tanase, C., Merialdo, B., Farinella, G., and Li, Q. 2011. "Eurecom at trecvid 2011: The light semantic indexing task", In Proceedings of the 9th TRECVID Workshop, Gaithersburg, USA.

Nister, D., and Stewenius, H. 2006. "Scalable recognition with a vocabulary tree", In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Vol. 2, CVPR '06, pp. 2161–2168, Washington, DC, USA. IEEE Computer Society.

Okamoto, H., Yasugi, Y., Babaguchi, N., and Kitahashi, T. 2002. "Video clustering using spatiotemporal image with fixed length", In Proceedings of the 2002 IEEE International Conference on Multimedia and Expo, ICME '02, vol. 1, pp. 53–56.

Papineni, K., Roukos, S., Ward, T., Zhu, W. 2002. "BLEU: a method for automatic evaluation of machine translation", In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 311-318.

Pass, G., Zabih, R., and Miller, J. 1996. "Comparing images using color coherence vectors", In Proceedings of the fourth ACM international conference on Multimedia, MULTIMEDIA '96, pp. 65–73, New York, NY, USA. ACM.

Pei, S.-C., and Chou, Y.-Z. 2002. "Effective wipe detection in mpeg compressed video using macro block type information", IEEE Transactions on Multimedia, vol. 4(3), pp. 309–319.

Perronnin, F., Sanchez, J., Mensink, T. 2010. "Improving the Fisher kernel for large-scale image classification", In: 11th Eur. Conf. on Computer Vision: Part IV. pp. 143-156. Springer-Verlag.

Petersohn, C. 2008. "Logical unit and scene detection: a comparative survey", In Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, volume 6820 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series.

Piskorski, J., Yangarber, R. 2013. "Information Extraction: Past, Present and Future", In: Poibeau T. et al. (Eds.), Multi-source, Multilingual Information Extraction and Summarization, Springer series: Intelligent Systems Reference Library, Vol. 42, Springer, pp. 23-49.

Qiu, G. 2002. "Indexing chromatic and achromatic patterns for content-based colour image retrieval", Pattern Recognition 35, pp. 1675-1686.

Rabiner, L. R. 1989. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE, Band 77, Nr. 2, pp. 257–286.

Rasheed, Z., and Shah, M. 2003. "Scene detection in Hollywood movies and TV shows", In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 343–348.

Rasheed, Z., and Shah, M. 2005. "Detection and representation of scenes in videos", IEEE Transactions on Multimedia, vol. 7(6), pp. 1097–1105.

Roberts, A., Gaizauskas, R., Hepple, M, Guo, Y. 2008. "Combining terminology resources and statistical methods for entity recognition: an evaluation", Proc. LREC Marrakech.

Rybach, D., Gollan, C., Heigold, G., Hoffmeister, B., Löff, J., Schlüter, R., and Ney, H. 2009. "The RWTH Aachen University Open Source Speech Recognition System", In Interspeech, pp. 2111-2114, Brighton, UK.

STATISTICA, Creators of STATISTICA Data Analysis Software and Services: Support Vector Machines (SVM), <http://www.statsoft.com/textbook/support-vector-machines/>.

Sande, K.E.A., Gevers, T., and Snoek, C.G.M. 2010. "Evaluating color descriptors for object and scene recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32(9), pp. 1582–1596.

Saux, B., and Amato, G. 2006. "Image classifiers for scene analysis", In: K. Wojciechowski, B. Smolka, H. Palus, R.S. Kozera, W. Skarbek, and L. Noakes, editors, *Computer Vision and Graphics*, vol. 32 of *Computational Imaging and Vision*, pp. 39–44. Springer Netherlands.

Seymore, K. McCallum, A., and Rosenfeld, R. 1999. "Learning hidden Markov model structure for information extraction", In *Proceedings of AAAI 99. Workshop on Machine Learning for Information Extraction*, pp. 37–42.

Shotton, J., Johnson, M., and Cipolla, R. 2008. "Semantic texton forests for image categorization and segmentation", In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR '08*, pp. 1–8.

Sidiropoulos, P., Mezaris, V., Kompatsiaris, I., Meinedo, H., Bugalho, M., and Trancoso, I. 2011. "Temporal video segmentation to scenes using high-level audiovisual features", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21(8), pp. 1163–1177.

Smeaton, A.F., Over, P., and Kraaij, W. 2009. "High-Level Feature Detection from Video in TRECVID: a 5-Year Retrospective of Achievements", In Ajay Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*, pp. 151–174. Springer Verlag, Berlin.

Snoek, C.G.M., Sande, K.E.A., Li, X., Mazloom, M., Jiang, Y.-G., Koelma, D.C., and Smeulders, A.W.M. 2011. "The MediaMill TRECVID 2011 semantic video search engine", In *Proceedings of the 9th TRECVID Workshop*, Gaithersburg, USA.

Souvannavong, F., Merialdo, B., and Huet, B. (2005): "Region-based video content indexing and retrieval", In *Fourth International Workshop on Content-Based Multimedia Indexing, CBMI '05*, pp. 21–23.

Spyrou, E., Tolia, G., Mylonas, P., and Avrithis, Y., 2009. "Concept detection and keyframe extraction using a visual thesaurus", *Multimedia Tools Appl.*, vol. 41(3), pp. 337–373.

Stein, D., Apostolidis, E., Mezaris, V., de Abreu Pereira, N., Müller, J., Sahuguet, M., Huet, B., and Lašek, I. 2012b. "Enrichment of news show videos with multimodal semi-automatic analysis", In *NEM-Summit*, Istanbul, Turkey.

Stein, D., Apostolidis, E., Mezaris, V., de Abreu Pereira, N., and Müller, J. 2012a. "Semi-automatic video analysis for linking television to the web", In *Proc. FutureTV Workshop*, pp. 1–8, Berlin, Germany.

Stein, D., Apostolidis, E., Sidiropoulos, Gkalelis, N., Mezaris, V., Sahuguet, M., Huet, B., Lašek, I., Kliegr, T. 2013. "D1.2 - Visual, text and audio information analysis for hypervideo, first release", Public Deliverable, The LinkedTV Project (FP7-ICT-2011-7), <http://www.slideshare.net/linkedtv/visual-text-and-audio-information-analysis-for-hypervideo-first-release>.

Steinberger, R. 2010. "Challenges and solutions for multilingual text mining", *Proc LREC Malta*.

- Stolcke, A. 2002. "SRILM - An Extensible Language Modeling Toolkit", In Proc. Int. Conf. on Spoken Language Processing, Denver, CA, USA.
- Sugano, M., Hoashi, K., Matsumoto, K., and Nakajima, Y. 2004. "Shot boundary determination on MPEG compressed domain and story segmentation experiments for trecvid 2004, in trec video retrieval evaluation forum", In Proceedings of the TREC Video Retrieval Evaluation (TRECVID). Washington D.C.: NIST, pp. 109–120.
- Tahir, M. A., Yan, F., Barnard, M., Awais, M., Mikolajczyk, K., and Kittler, J. 2010. "The university of surrey visual concept detection system at imageCLEF@ICPR: working notes", In Proceedings of the 20th International conference on Recognizing patterns in signals, speech, images, and videos, ICPR'10, pp. 162–170, Berlin, Heidelberg, 2010. Springer-Verlag.
- Tan, W., Teng, S., and Zhang, W. 2007. "Research on video segmentation via active learning", In Proceedings of the Fourth International Conference on Image and Graphics, ICIG '07, pp. 395–400, Washington, DC, USA, IEEE Computer Society.
- Tao, R. 2011. "Visual concept detection and real time object detection", CoRR, abs/1104.0582.
- Thurmair, Gr., Aleksic, V. 2012. "Large-scale lexical analysis", Proc. LREC Istanbul.
- Tola, E., Lepetit, V., and Fua, P. 2008. "A fast local descriptor for dense matching", In Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8.
- Tola, E., Lepetit, V., and Fua, P. 2010. "Daisy: An efficient dense descriptor applied to widebaseline stereo", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32(5), pp. 815–830.
- Truong, B. T., Venkatesh, S., and Dorai, C. 2003. "Scene extraction in motion pictures", IEEE Transactions on Circuits and Systems for Video Technology, vol. 13(1), pp. 5–15.
- Tsamoura, E., Mezaris, V., and Kompatsiaris, I. 2008. "Gradual transition detection using color coherence and other criteria in a video shot meta-segmentation framework", In Image Processing. ICIP (2008): 15th IEEE International Conference on, pp. 45–48.
- Tuytelaars, T., and Mikolajczyk, K. 2008. "Local Invariant Feature Detectors: A Survey", Now Publishers Inc., Hanover, MA, USA.
- Tyers, F.M., and Alperen, M. 2010. "South-East European Times: A parallel corpus of the Balkan languages", In Proceedings of the Workshop on Exploitation of multilingual resources and tools for Central and (South) Eastern European Languages.
- Van de Sande, K., Gevers, T., and Snoek, C. 2008. "A comparison of color features for visual concept classification", In Proceedings of the 2008 international conference on Content-based image and video retrieval, CIVR '08, pp. 141–150, New York, NY, USA. ACM.
- Van de Sande, K., Gevers, T., and Snoek, C. 2010a. "Evaluating color descriptors for object and scene recognition", IEEE Trans. Pattern Anal. Mach. Intell., vol. 32(9), pp. 1582–1596.
- Van de Sande, K.E.A., Snoek, C.G.M., Smeulders, A.W.M. 2014. "Fisher and vlad with fair", In: IEEE Conference on Computer Vision and Pattern Recognition.

- Van de Weijer, J., Gevers, T., and Bagdanov, A.D. 2006. "Boosting color saliency in image feature detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28(1), pp. 150–156.
- Vedaldi, A., and Fulkerson, B. 2010. "Vlfeat: an open and portable library of computer vision algorithms", In *Proceedings of the international conference on Multimedia (MM '10)*. ACM, New York, NY, USA, pp. 1469-1472.
- Wang, J., and Chua, T.-S. 2002. "A framework for video scene boundary detection", In *Proceedings of the 10th ACM international conference on Multimedia, MULTIMEDIA '02*, pp. 243–246, New York, NY, USA. ACM.
- Wei, A., Pei, Y., and Zha, H. 2012. "Random-sampling-based spatial-temporal feature for consumer video concept classification", In *Proc. of the 2012 IEEE International Conference on Image Processing (ICIP)*, pp. 1861–1864.
- Willamowski, J., Arregui, D., Csurka, G., Dance, C. R., and Fan, L. 2004. "Categorizing nine visual classes using local appearance descriptors", In *ICPR Workshop on Learning for Adaptable Visual Systems*.
- Winograd, T. 1986. *Language as a Cognitive Process*, vol. 1 Syntax. Addison Wesley.
- Witten, I., Frank, E. 2005. "Data Mining Practical Machine Learning Tools and Techniques", Morgan Kaufmann, San Francisco, 2nd ed.
- Xie, L., Xu, P., Chang, S.-F., Divakaran, A., and Sun, H. 2004. "Structure analysis of soccer video with domain knowledge and hidden Markov models", *Pattern Recogn. Lett.*, vol. 25(7), pp. 767–775.
- Yeung, M., Yeo, B.-L., and Liu, B. 1998. "Segmentation of video by clustering and graph analysis", *Computer Vision Image Understanding*, vol. 71(1), pp. 94–109.
- Zabih, R., Miller, J., and Mai, K. 1999. "A feature-based algorithm for detecting and classifying production effects", *Multimedia Syst.*, vol. 7(2), pp. 119–128.
- Zhai, Y., and Shah, M. 2006. "Video scene segmentation using markov chain monte carlo", *IEEE Transactions on Multimedia*, vol. 8(4), pp. 686–697.
- Zhang, D.-Q., Lin, C.-Y., Chang, S.-F., and Smith, J.R. 2004. "Semantic video clustering across sources using bipartite spectral clustering", In *IEEE International Conference on Multimedia and Expo, ICME '04*, vol. 1, pp. 117 – 120.
- Zhang, J., Marszalek, M., Lazebnik, S., and Schmid, C. 2007. "Local features and kernels for classification of texture and object categories: A comprehensive study", *Int. J. Comput. Vision*, vol. 73(2), pp. 213–238.
- Zhu, J., Hoi, S. C. H., Lyu, M. R., and Yan, S. 2008. "Near-duplicate keyframe retrieval by nonrigid image matching", In *Proceedings of the 16th ACM international conference on Multimedia, MM '08*, pp. 41–50, New York, NY, USA. ACM.
- Zhu, S. and Liu, Y. 2009. "Video scene segmentation and semantic representation using a novel scheme", *Multimedia Tools and Applications*, vol. 42(2), pp. 183–205.

A Appendix: Annotation Guidelines

A.1 Introduction

These guidelines describe the annotation of texts with semantic data and associated lexical information. The annotation has the goal of facilitating the development and evaluation of advanced Information Extraction (IE) and Automatic Summarisation (AS) techniques. More precisely, the guidelines describe how to identify and mark in the text facts involving entities potentially found in some target datasets containing factual data, and how to enrich the annotations with lexical information pertinent to the linguistic expressions that indicate the facts in the text. The datasets we consider are linked open datasets (LOD), while the lexical information is obtained from freely available online thesauri and lexical indexes, some of them also published as LOD.

The annotation procedure is divided into two separate annotation subtasks. The first subtask consists in marking nominal expressions found in lexical resources and that may denote entities in the target datasets. This is similar to Named Entity Disambiguation (NED)⁵⁵, a task that various tools address automatically with varying levels of success. We look up the nominal expressions in the lexical resources and, for each of these resources, record whether the expression was found. The second subtask is to mark verbal predicates that indicate events or other relations involving one or more of the annotated entities. Predicates are also looked up in online lexical resources and their presence in each resource is annotated. In addition, the information found in the lexical entries is used to determine and annotate what entities or other relations act as arguments of the predicate.

The guidelines have been developed during an experimental manual annotation of journalistic texts about energy policies. As a result, they are illustrated with examples of journalistic texts about energy policies. The guidelines, however, are valid for the annotation of texts of any genre and domain with any lexical resources and target datasets.

The following resources were used in the experimental annotation:

- Lexical resources:
 1. BabelNet⁵⁶: Index of mapped entries in Wordnet, Open Multilingual Wordnet, Wikipedia, DBPedia, Wiktionary and others.
 2. Unified Verb Index⁵⁷: Index of mapped lexical entries in predicate dictionaries (VerbNet, PropBank) and predicative sense dictionary (FrameNet).

⁵⁵ Also known as entity linking, see http://en.wikipedia.org/wiki/Entity_linking

⁵⁶ See <http://babelnet.org/search.jsp>

⁵⁷ See <http://verbs.colorado.edu/verb-index/search.php>

3. Eurovoc⁵⁸: multilingual thesaurus containing terms obtained from official documents of the European Parliament.
4. Reegle Glossary⁵⁹: multilingual thesaurus containing terms belonging to the renewable energies domain.

2 Factual datasets:

1. Wikipedia/DBPedia⁶⁰: DBPedia is an RDF dataset with data extracted from semi-structured data in Wikipedia pages. Most pages in Wikipedia have a corresponding entry in DBpedia (a dereferencable URI-named entity).
2. Reegle data⁶¹: a dataset containing the terms from the Reegle Glossary plus a list of important actors (organisations and individuals) for the renewable domain.

A.2 Annotation of nominal expressions

1. Annotate nominal expressions with the annotation type 'Entity'.
 - Annotate dates, numbers and quantifiers as separate entities, *e.g.* '2014', 'Monday', '50%', 'most', 'two'.
 - Exclude determiners from the annotation, *e.g.* in 'the Fukushima disaster' annotate 'Fukushima disaster'.
 - Do not annotate possessives, rhetorical expressions or idioms. *e.g.* 'to say the least'.
2. In the case of complex nominal expressions, annotate the whole expression as a single entity if it has an entry in the target resources:
 - Bablenet
 - Eurovoc
 - Reegle
 - *E.g.* if 'wind farms' has an entry in any of BableNet, Eurovoc and Reegle, annotate 'wind farms' and do not annotate 'wind' nor 'farms'. However, if there is no entry for the whole expression in any of them, annotate 'wind' and 'farms' as separate entities (provided they do have entries).
3. For each annotation of type 'Entity', except for dates, numbers and quantifiers, complete its annotation with the following:
 - *BabelNet*: How is it found in BabelNet?

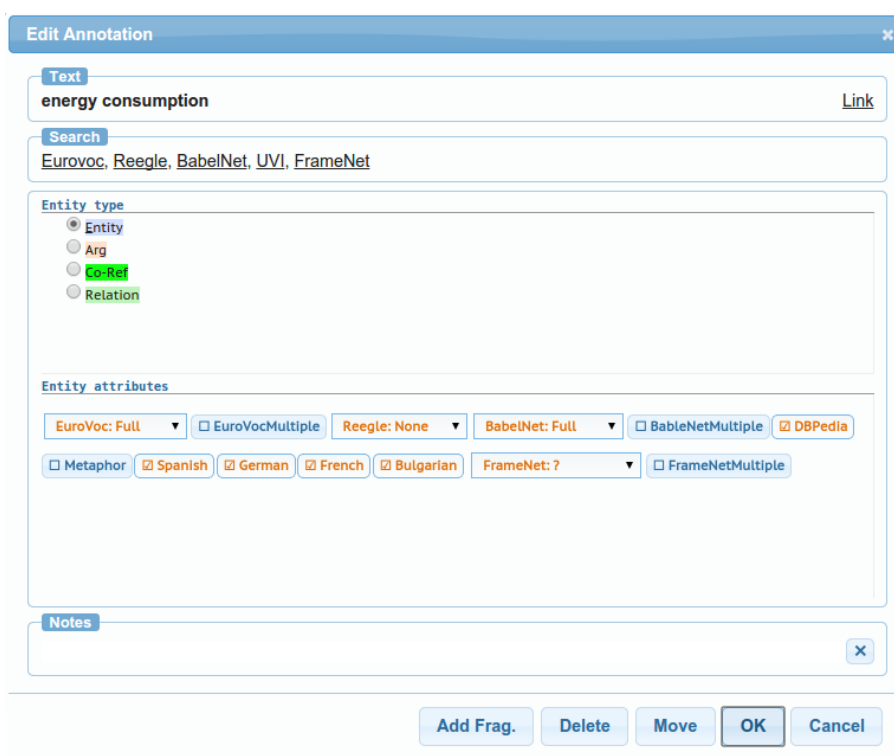
⁵⁸ See <http://eurovoc.europa.eu/drupal/>

⁵⁹ See <http://www.reegle.info/glossary>

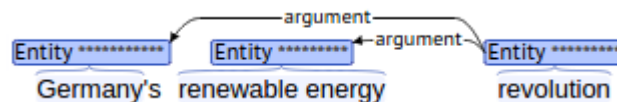
⁶⁰ See <http://dbpedia.org>

⁶¹ See <http://www.reegle.info/index.php>

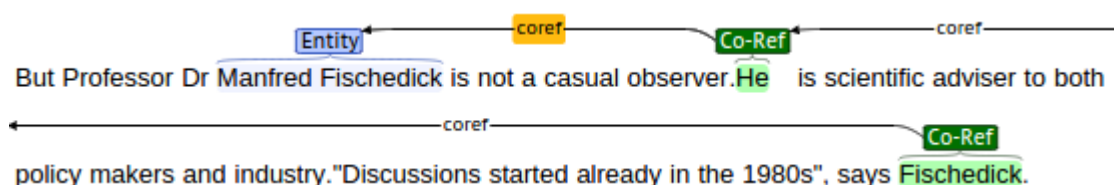
- *BabelNetMultiple*: Does the nominal expression lead to multiple senses in BabelNet?
- *EuroVoc*: How is it found in EuroVoc?
- *EuroVocMultiple*: Does the nominal expression lead to multiple senses in EuroVoc?
- *Reegle*: How is it found in Reegle?
- Each of the above questions can take one of the following answers:
 - i. *full*, if entity found in the resource using nominal expression exactly as it is (ignoring inflection).
 - ii. *partial*, if the entity is found in the resource by either adding or dropping some words from the nominal expression.
 - iii. *none* if not found in the resource *using the nominal expression (includes the case when it may be found using synonyms or equivalent expressions)*.
- *Metaphor*: Does the nominal expression have a metaphorical meaning?
- *Spanish, German, French, and Bulgarian*: mark languages for which there are equivalent expressions in BabelNet, EuroVoc or Reegle.
- *Notes*: write down in the notes section of the form the BabelNet sense (e.g. populace¹) and EuroVoc exact term ('nuclear power' leads to 'nuclear industry' in EuroVoc).



4. Add an 'argument' relation from the 'Entity' annotation of a nominal expression and the 'Entity' annotations of nominal expression it is modified by.



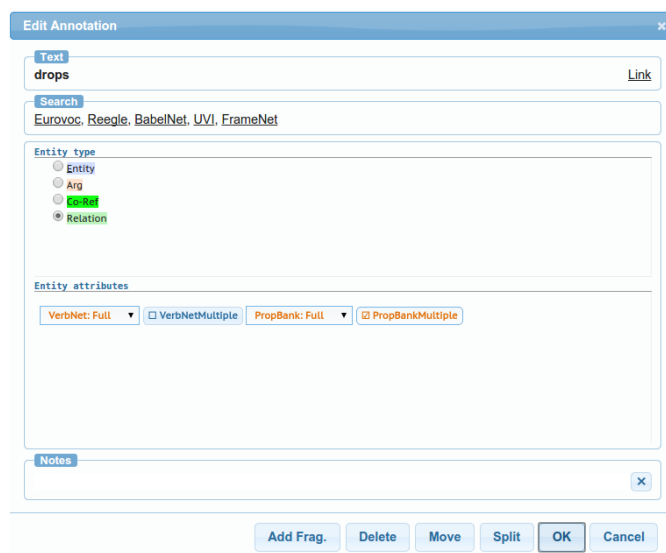
5. If an entity has arguments, complete its annotation with the following:
 - *FrameNet*: Is the predicative noun found as a lexical unit in FrameNet?
 - *FrameNetMultiple*: Is the lexical unit linked to multiple frames?
 - If there is a frame for the sense of the nominal predicate in FrameNet, write down the type (e.g. food, support, etc.) of each core or non-core frame element in the *Notes* section of the corresponding 'argument' relation.
6. In the case of anaphoric and cataphoric expressions (pronouns and other pro-forms) and co-referring repetitions of exactly the same expression, mark them with the annotation type 'Co-ref'.
 - Link the new 'Co-ref' annotation to the previous mention using the relation 'coref'.
 - Also mark cases of multiple or partial co-reference, e.g. '**John** and **Peter** like to walk alone. **They** often do'.
7. If a non-anaphoric nominal mention can be inferred to co-refer with a preceding and non-identical nominal expression, add a 'coref' relation between the two.
 - Also mark cases of multiple or partial co-reference, e.g. '**Germany** and **France** oppose the new legislation. **Both countries** demand changes.'



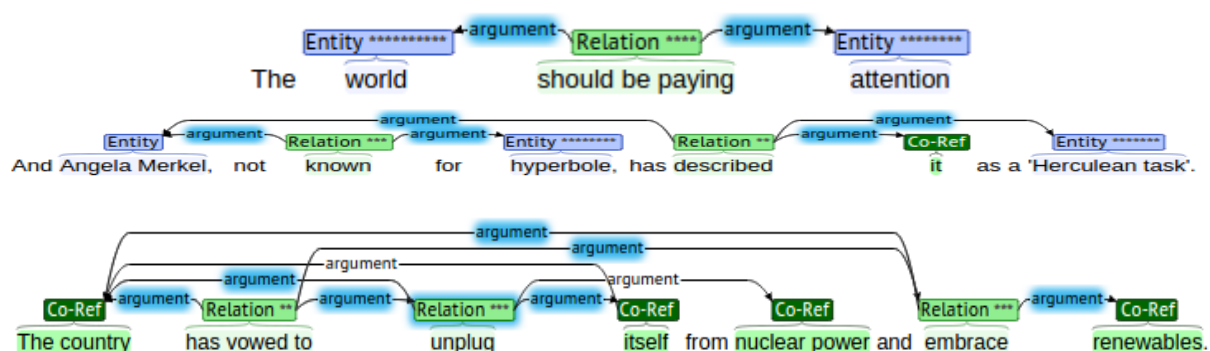
A.3 Annotation of verbal expressions

1. Annotate all verbs with the annotation type 'Relation'
 - Include support verbs in the annotation, e.g. '*may have been passed*'.
 - Include prepositions in phrasal verbs, e.g. '*shut down*',
 - Exclude adverbs and prepositions introducing arguments, e.g. *in* '*even has*' and '*not known*' do not annotate '*even*' neither '*not*'.
 - Do not attempt the annotation of elided verbs.
2. For each verb annotated, complete its annotation with the following:
 - *PropBank*: Is the predicate found in PropBank?
 - *PropBankMultiple*: Does it have multiple senses?
 - *VerbNet*: Do the arguments of the PropBank sense have mappings to VerbNet roles?
 - *VerbNetMultiple*: Are there multiple VerbNet roles for one or more of the PropBank arguments?

- *FrameNet*: Is the verb found as a lexical unit in FrameNet?
- *FrameNetMultiple*: Is the corresponding lexical unit in FrameNet mapped to multiple frames?
- *Notes*: write down in the notes section of the form the PropBank sense (e.g. unplug.01) and FrameNet frame (e.g. Commitment).



- Link relation annotations to their participants using the relation 'argument'.
 - Link the annotation 'Relation' of the verb to the annotated core of its argument or modifier, which can be an annotation of type 'Entity', 'Co-Ref' or 'Relation'.
 - Do not annotate prepositions used to introduce arguments, e.g. in 'from nuclear power' do not annotate 'from' as part of the verb nor the argument.



- Write down in the 'Notes' section of the form of each 'argument' relation:
 - If the argument corresponds to a core argument according to PropBank, write down its label (e.g. Arg1-PPT)
 - If the argument corresponds to any core or non-core frame element in FrameNet, write down the element type (e.g. food, support, etc.).

Edit Annotation

From

Relation ("ensure")

Link

To

Relation ("drops")

Type

☒ argument

Notes

PB= Arg1-PPT VN=Theme

Reverse

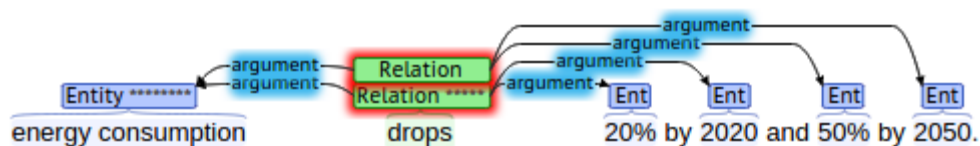
Delete

Reselect

OK

Cancel

- In coordinations of arguments, annotate the verbal relation once for each coordination member, and link each coordinated argument to one of the 'Relation' annotations.



B Appendix: Named Entities Recognition - Specification

B.1 Entity types

The NE recogniser will try to identify the following entity types; it will also give details for the respective types in the 'attributes' fields:

Persons

Persons can be described by first and lastname, and in some cases, gender and occupation may be recognised. This information is stored in attributes together with the person name.

Locations

The most important and most frequent locations are countries and cities. We also investigate regions ('Auvergne', including US. state names like 'Indiana') and street names. All other location names come as 'other' (including continents like 'Europe', or names like 'Kremlin', 'White House' etc.).

Organisations

Organisations can be companies ('Apple') or institutions ('European Commission') or other private entities (like 'FC Everton', 'Cosa Nostra', 'Greenpeace').

Products and brand names

This is for products like 'Nivea', 'BMW 317i', 'Miele 400'. Products are difficult to distinguish from company names by formal means: 'Maytag' can be a product / brand name as well as a company. In such cases, ambiguous results may be delivered.

Time expressions

Time expressions are dates, partial dates (like 'November 1993' or 'November 16th'), hour descriptions ('15:00 CET'). Also relational information 'three days ago', 'on Wednesday' is stored, in case someone can make use of it. Time expressions set their respective attributes ('year', 'month' etc.)

Amounts

Units consist of an amount (25) and a unit (km, €, etc.), or a percentage. The most important units are prices.

Communication means

This type identifies URLs, email addresses, phone and fax numbers, and returns the respective type attribute.

Names

There are many other named entities not falling into one of the types above, like the 'Maytag Repairman' or the 'Clear Water Act', 'European Monetary System' and so on. They are collected here under just 'Name'.

B.2 Objects and attributes

The JSON serialisation of the NER will provide the following objects and attributes:

B.2.1 Document

This is the top level object. One document is one object. It has three attributes

Attributes of document are:

"LANGUAGE":	"<lg>", the language of the document
"TEXT":	"<text>", the text of the document {quotes to be escaped}
"SENTENCES":	give the sentence segmentation of the text, as a list of triplets of <"text" , "onset" , "offset">
"TOKENS":	give the token sequence of the text, as a list of triplets of <"text" , "onset" , "offset">
"ENTITIES":	gives all named entities found in the document. Details are described below.

Other document-related attributes could be: encoding (we assume everything is in UTF8 no-BOM); topic (assigned by a topic identifier component), and other attributes set by the following processing components.

Input of the component is pure text, and a language attribute. The output structure may not have an 'ENTITIES' attribute (depending on the presence of NEs in the text).

B.2.2 Entities

The entities are sorted by the type of an entity. Supported entity types are the following:

"ENTITIES":

{	
"PERSONS": []	a list of all persons
"LOCATIONS": []	a list of all locations
"ORGANISATIONS": []	a list of all organisations
"BRANDS": []	a list of all brands / products
"TIMES": []	a list of all time expresions
"AMOUNTS": []	a list of all units
"COMMUNICATIONS": []	a list of all communication means
"-names": []	a list of all non-typed entities
}	

These entities are described in more detail below. All type sections are optional; i.e. if a document does not contain any brand, for instance, the respective attribute is not offered.

B.2.3 Entity type structures

The entity types just mentioned have a uniform internal structure, consisting of the following attributes:

"name": "<a canonical name>",	the name in its 'basic' form
"count": "<#-occurrences>",	how often this entity occurs in the document
"url": "<an url>"	pointer to an URI for this entity (e.g. DBPedia)
"attributes": { }	the attributes for this entity type, cf. below
"matches": []	the single matches of the entity type in the input

The different entity types differ only in their set of attributes.

Names

As for the names themselves, the canonical form of the name are built as follows: creation of standard casing ('LONDON' -> 'London'), removal of all case attributes ('Germany's' -> 'Germany'), longest string is kept, aside from company designators ('Handelsblatt' / 'Handelsblatt GmbH' / 'Handelsblatt GmbH & Ko KG' -> 'Handelsblatt'). For dates, the input form is left as canonical form in V1.0; in later versions, the ISO normalisation can be used. Units stay in their input form; details are given in their attributes. Person names come in the longest form found.

Attributes

The attributes object contains different attributes depending on the entity type. The following attributes will be tried to be identified. All attributes are optional. They depend on the document text and the available resources:

Person

'Person' has the following attributes:

"firstname": "<string>",	the firstname
"lastname": "<string>",	the lastname
"gender": "<member>",	the gender if available. Values are: {masculine, feminine}
"occupation": "<string>"	the occupation if available (not in V1)

(We will not keep middle names, titles etc. in attributes. Middle names may occur in the normalised form of the name, titles will be excluded).

Location

'Location' has the following attributes:

"type": <member>	Values are: {country, region, city, street, other}
------------------	--

The value of 'other' can include nature things (mountains, continents, islands), buildings, and so on; they are not specified further.

Organisation

'Organisation' has the following attributes:

"type": <member>	Values are: {company, institution}
------------------	------------------------------------

Companies refer to businesses. By 'institutions', mainly governmental and other public law organisations ('Red Cross') will be marked, but also all kinds of football clubs, mafia, 'Translators without Boundaries' etc. are covered here.

Brand / Product

'Product' has the following attributes: (none in V1.0)

// "type": "<String>" (don't know yet what makes sense here)

(We need more experience with this type of NE here).

Times

'Time' has the following attributes:

"year": "<string>",	a year
"month": "<string>",	a month, in canonical notation (digit)
"day": "<string>",	a day
"time": "<string>",	an hour ('15:00 CET')
"weekday": "<string>"	a weekday ('Friday'); could also go under 'rel'
"rel": "<string>"	a relative expression ('the last three days')
"other": "<string>"	any other time expression ('Valentine's day')

Note that many cases of partial filling exist ("in November 1946": no day).

Amounts

'Amounts' have the following attributes:

"unit": "<string>"	a unit designator, or "%"
"amount": "<number>"	an amount
"type": "<member>"	a type. One of {unit, price}

The type attribute, for the time being, would just differentiate between prices and other amounts to ease access to prices. Canonical values for prices / currencies are the international abbreviations ('EUR').

Communications

'Communications' has the following attributes:

"type": "<string>"	the type of communication means
--------------------	---------------------------------

There will be {URL, email, fonfax} as values for this type feature.

Names

'Name' has the following attributes: (none in V1.0))

// "type": "<string>"	a type if anything can be identified, otherwise empty
-----------------------	---

This for NEs which cannot be typed, like 'Clean Water Act', 'Maytag Repairman' and others.

Matches

This a list of objects describing where the entity occurred in the text. Attributes are:

"text": "<string>",	the text form of this entity
"onset": "<number>",	the start of the NE

"offset": <number>"

the end of the NE

The objects can have different text forms (like *'Austria'* and *'Austria's'*, or *'Ford Corp.'* and *'Ford'*), therefore the text form is stored in the *'matches'*, and the object has a canonical form of the name. Also, as the text forms can have different length, we give start and end of the offset.

B.3 Output example as JSON object

```
{
  "document": {
    "LANGUAGE": "en",
    "TEXT": "We met Peter Brady in San Prodotto. He works for Rusticon Srl. Brady explained that Rusticon exports spoons and sporks. ",
    "SENTENCES":
    [
      { "text": "We met Peter Brady in San Prodotto.", "onset": "0", "offset": "35" },
      { "text": "He works for Rusticon Srl.", "onset": "36", "offset": "62" },
      { "text": "Brady explained that Rusticon exports spoons and sporks.", "onset": "63", "offset": "119" }
    ],
    "TOKENS":
    [
      { "text": "We", "onset": "0", "offset": "2" },
      { "text": "met", "onset": "3", "offset": "6" },
      { "text": "Peter", "onset": "7", "offset": "12" },
      { "text": "Brady", "onset": "13", "offset": "18" },
      { "text": "in", "onset": "19", "offset": "21" },
      { "text": "San", "onset": "22", "offset": "25" },
      { "text": "Prodotto", "onset": "26", "offset": "34" },
      { "text": ".", "onset": "34", "offset": "35" },
      { "text": "He", "onset": "36", "offset": "38" },
      { "text": "works", "onset": "39", "offset": "44" },
      { "text": "for", "onset": "45", "offset": "48" },
      { "text": "Rusticon", "onset": "49", "offset": "57" },
      { "text": "Srl", "onset": "58", "offset": "61" },
      { "text": ".", "onset": "61", "offset": "62" },
      { "text": "Brady", "onset": "63", "offset": "68" },
      { "text": "explained", "onset": "69", "offset": "78" },
      { "text": "that", "onset": "79", "offset": "83" },
      { "text": "Rusticon", "onset": "84", "offset": "92" },
      { "text": "exports", "onset": "93", "offset": "100" },
      { "text": "spoons", "onset": "101", "offset": "107" },
      { "text": "and", "onset": "108", "offset": "111" },
      { "text": "sporks", "onset": "112", "offset": "118" },
      { "text": ".", "onset": "118", "offset": "119" }
    ],
    "ENTITIES": {
      "PERSONS":
      [
        {
          "name": "Peter Brady",
          "count": "2",
          "url": "www.dummy.com",
          "attributes":

```

```
        { "gender": "male" },
    "matches":
    [
        { "text": "Peter Brady", "onset": "7", "offset": "18" },
        { "text": "Brady", "onset": "63", "offset": "68" }
    ]
    },
],
"LOCATIONS":
[
    {
        "name": "San Prodotto",
        "count": "1",
        "url": "www.dummy.com",
        "attributes":
            { "type": "city" },
        "matches":
        [
            { "text": "San Prodotto", "onset": "22", "offset": "34" }
        ]
    }
],
"ORGANISATIONS":
[
    {
        "name": "Rusticon",
        "count": "2",
        "url": "www.dummy.com",
        "attributes":
            { "type": "company" },
        "matches":
        [
            { "text": "Rusticon", "onset": "49", "offset": "57" },
            { "text": "Rusticon", "onset": "84", "offset": "92" }
        ]
    }
]
}
}
}
```

C Appendix: Techniques for multimedia concept detection

C.1 Creation of training dataset

For both APIs, we have obtained an identification key and password. Regarding the number of queries that can be sent to the APIs, the Bing API provides for free the results of 5000 transactions per month. Flickr API on the other hand does not impose any constraints, but it should be noted that the quality of the images returned in terms of relativity with the query is significantly lower. Both APIs return a set of image URLs that are considered relevant to the query. Finally, as far as the Google Images is concerned, there is no specific number of queries that can be sent within specific time period; however we inserted a delay in the queries in order to avoid being considered a robot. Then, the result page is downloaded and parsed and the included URLs of the images are retrieved. It should be noted that at a later stage, we plan on using the Google API as well. In the sequel, the lists of the image URLs retrieved by the three sources are merged and the images are downloaded.

C.2 Evaluation

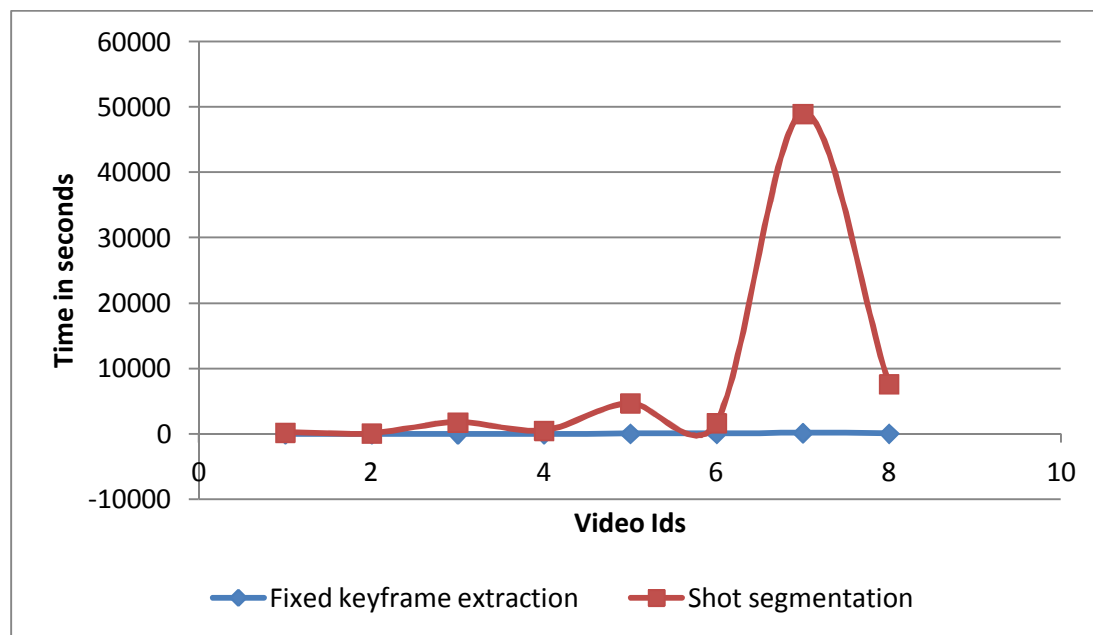


Figure C1: Keyframe extraction methods compared in time.

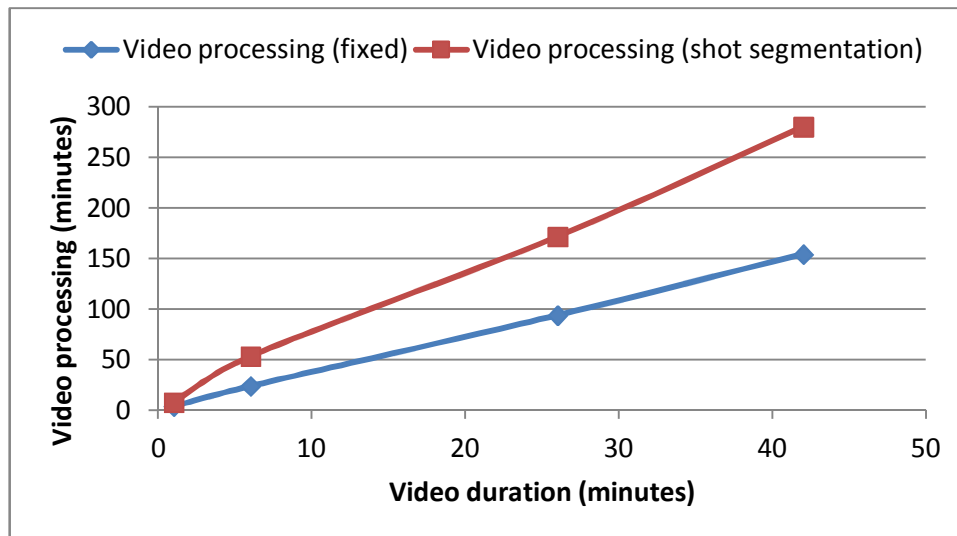


Figure C2: Time required for video processing for videos with different durations.

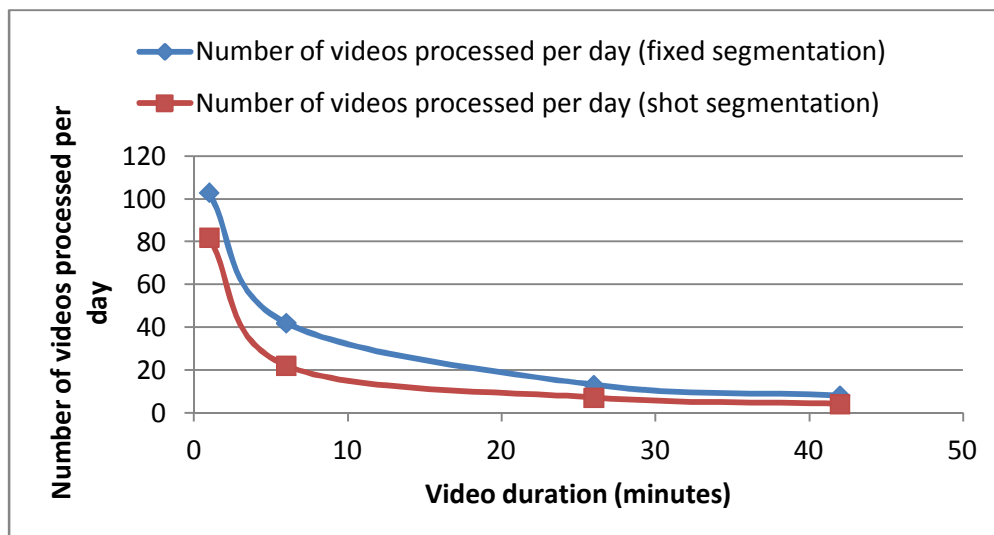


Figure C3: Number of videos processed from one PC in one day for video with different durations.