

MULTISENSOR

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

FP7-610411

D4.3

Multimodal indexing module for topic detection and retrieval

Dissemination level:	Public
Contractual date of delivery:	Month 28, 29/02/2016
Actual date of delivery:	Month 28, 29/02/2016
Workpackage:	WP4 Multidimensional content integration and retrieval
Task:	T4.1 Topic-based modelling T4.4 Multimodal indexing and retrieval
Type:	Prototype
Approval Status:	Final Draft
Version:	1.0
Number of pages:	64
Filename:	D4.3_MultimodalIndexing_2016-02-29_v1.0.docx

Abstract

The document describes the techniques for multimodal topic detection and retrieval of socially interconnected multimedia-enriched objects (SIMMO). Topic detection is approached as news clustering, where the number of topics is efficiently estimated and multimedia retrieval is based on multimodal fusion of textual and visual features. Finally, the advanced multimodal category-based classification module is presented.

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	16/01/2016	Document structure	D. Liparas, S. Vrochidis (CERTH)
0.2	18/02/2016	Contributions	I. Gialampoukidis, A. Moumtzidou, T. Mironidis, D. Liparas (CERTH)
0.3	24/02/2016	Integrated document	D. Liparas (CERTH)
0.4	24/02/2016	Review of CERTH contribution	S. Vrochidis (CERTH)
0.5	24/02/2016	Internal Review of the whole document	S. Mille (UPF)
1.0	29/02/2016	Final version	D. Liparas (CERTH)

Author list

Organization	Name	Contact Information
CERTH	Dimitris Liparas	dliparas@iti.gr
CERTH	Ilias Gialampoukidis	heliassgj@iti.gr
CERTH	Anastasia Moumtzidou	moumtzid@iti.gr
CERTH	Theodoros Mironidis	mironidis@iti.gr
CERTH	Stefanos Vrochidis	stefanos@iti.gr

Executive Summary

This deliverable presents the final version of the topic-based modelling and the multimodal indexing and retrieval modules.

Specifically, D4.3 reports on the results of Task 4.4 (Multimodal indexing and retrieval), with respect to the SIMMO multimedia indexing model, which was initially presented in a previous deliverable (D4.1), as well as the proposed approach for multimedia retrieval, which is based on multimodal fusion of textual and visual features. In addition, D4.3 presents the advanced techniques for topic detection (Task 4.1 – Topic-based modelling) based on multimodal clustering of News Items, along with an efficient methodology for the estimation of the number of topics in a dataset named DBSCAN-Martingale. Moreover, the updated and final multimodal framework for category-based classification (Task 4.1 – Topic-based modelling) is described. This framework relies on the Random Forests machine learning method and late fusion strategies that are based on the operational capabilities of Random Forests. The results of several experiments conducted within MULTISENSOR for the evaluation of all proposed frameworks (multimedia retrieval, topic detection and category-based classification) are presented and discussed in the current deliverable. Finally, the components and resources of the multimodal indexing and retrieval, category-based classification and topic detection modules are described in detail. D4.3 contributes to the achievement of Milestone MS5 (Final System).

Abbreviations and Acronyms

AP	Average Precision
BoW	Bag-of-Words
CBOW	Continuous Bag-of-Words
CEP	Content Extraction Pipeline
CNN	Convolutional Neural Networks
CNR	Central News Repository
DB	DataBase
DC	Document Classification
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
LBP	Local Binary Patterns
JSON	JavaScript Object Notation
KB	Knowledge Base
KNN	K Nearest Neighbour
LDA	Latent Dirichlet Allocation
MAP	Mean Average Precision
NLP	Natural Language Processing
NN	Neural Network
NNLM	Neural Net Language Model
NMI	Normalized Mutual Information
OOB	Out-Of-Bag
OPS	OPerationS repository
RF	Random Forests
PLS	Partial Least Squares
SIFT	Scale-invariant feature transform
SIMMO	Socially Interconnected MultiMedia-enriched Objects
SoA	state-of-the-art
SVM	Support Vector Machine
T	Tasks

Table of Contents

1	INTRODUCTION	8
1.1	Architecture	9
2	MULTIMODAL INDEXING AND RETRIEVAL.....	11
2.1	Relevant work on content-based multimedia information retrieval.....	12
2.2	The MULTISENSOR multimedia retrieval framework	13
2.2.1	Preliminaries	13
2.2.2	Multimedia retrieval using multiple modalities.....	14
2.3	Multimodal indexing and retrieval module	18
2.3.1	Multimodal indexing.....	18
2.3.2	Multimedia retrieval module.....	23
2.4	Application to MULTISENSOR Use Cases and evaluation.....	23
2.4.1	Evaluation data	23
2.4.2	Evaluation results.....	24
3	TOPIC-BASED CLASSIFICATION	27
3.1	The MULTISENSOR category-based classification framework based on multimodal features.....	28
3.2	Category-based classification module	31
3.3	Application to MULTISENSOR Use Cases and evaluation.....	31
3.3.1	Evaluation data	31
3.3.2	Evaluation results.....	33
4	TOPIC-EVENT DETECTION	37
4.1	Relevant work on topic-event detection and news clustering	38
4.2	The MULTISENSOR topic detection framework based on multimodal features.....	39
4.3	Estimation of the number of topics using the DBSCAN-Martingale	40
4.3.1	Notation and background on DBSCAN and OPTICS	40
4.3.2	Estimation of the number of clusters with the DBSCAN-Martingale.....	41
4.3.3	The martingale stochastic process.....	44
4.4	Topic detection module	45
4.5	Application to MULTISENSOR Use Cases and evaluation.....	46
4.5.1	Evaluation data	47

4.5.2	Evaluation results in public datasets	48
4.5.3	Evaluation results in the MULTISENSOR database	52
5	CONCLUSIONS	55
6	REFERENCES	56
A	APPENDIX	61
A.1	Confusion matrices (topic-based classification)	61
A.2	Estimation of the number of news clusters (topic-event detection)	64

1 INTRODUCTION

The current deliverable presents the work done with respect to tasks T4.1 (Topic-based modelling) and T4.4 (Multimodal indexing and retrieval) of Work Package 4 (WP4), during the time period M12-M28 of the MULTISENSOR project. T4.1 is divided into two subtasks, namely a) category-based classification and b) topic-event detection. On the other hand, T4.4 deals with the development of a multimodal indexing and retrieval module, based on a representation that is able to effectively capture enriched multimedia content.

Regarding task T4.1, D4.3 focuses on the topic-event detection subtask, since the main work on category-based classification has already been reported in deliverable D4.1 (submitted in M12). Specifically, topic-event detection is tackled as a clustering problem within MULTISENSOR and a hybrid clustering approach for assigning news articles into topics is proposed. In this approach, prior knowledge of the correct number of clusters/topics is not required, as this number is automatically estimated by means of a novel methodology. Moreover, the advancements made with respect to the topic-based classification framework described in D4.1 are reported and the final MULTISENSOR module for classifying news articles into generic categories is presented.

Regarding task T4.4, D4.3 presents the MULTISENSOR indexing and retrieval module, in which the advanced pyramidal vector-based representation, indexing and search techniques are described. Specifically, an efficient multimedia data representation framework, named Socially Interconnected MultiMedia-enriched Objects (SIMMO), is realized. In addition, the aforementioned module includes a multimedia retrieval approach that exploits multimodal fusion of textual and visual features.

The timeline of the two tasks along the lifetime of the project is depicted in Figure 1.



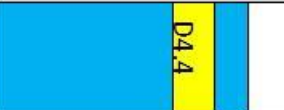
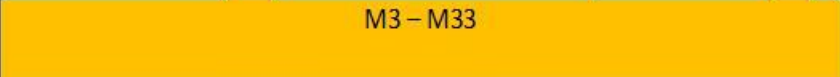
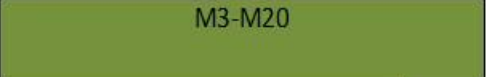
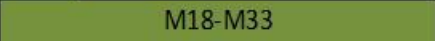
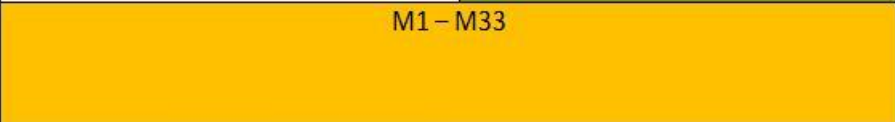
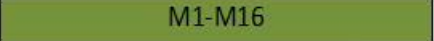
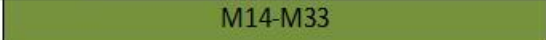
ACTIVITY	Y1	Y2	Y3
WP4			
T4.1 Topic-based modelling			
Category-based classification			
Topic-event detection			
T4.4 Multimodal Indexing and Retrieval			
Model development			
Indexing Structure			

Figure 1: Timeline of tasks T4.1 and T4.4

The rest of the document is organized as follows: Subsection 1.1 gives an overview of the discussed modules within the MULTISENSOR architecture. In Sections 2, 3 and 4 the

proposed frameworks for the multimodal indexing and retrieval, topic-based classification and topic-event detection modules, respectively, are presented. Finally, concluding remarks are provided in Section 5.

1.1 Architecture

This Subsection describes how the multimodal indexing and retrieval, topic-based classification and topic-event detection modules are integrated into the MULTISENSOR architecture.

Figure 2 depicts the Content Extraction Pipeline (CEP), as it was formed in the Second Prototype of the MULTISENSOR platform (see Section 3.3.3 – D7.6), with the topic-based classification module highlighted. The CEP contains modules (e.g. named entities recognition, dependency parsing, sentiment analysis etc.) that process and analyse the News Items stored in the Central News Repository (CNR). Before the processed News Items are stored into the Knowledge Base (KB) of the MULTISENSOR platform, the topic-based classification module assigns a category to each News Item (based on a predefined list of categories).

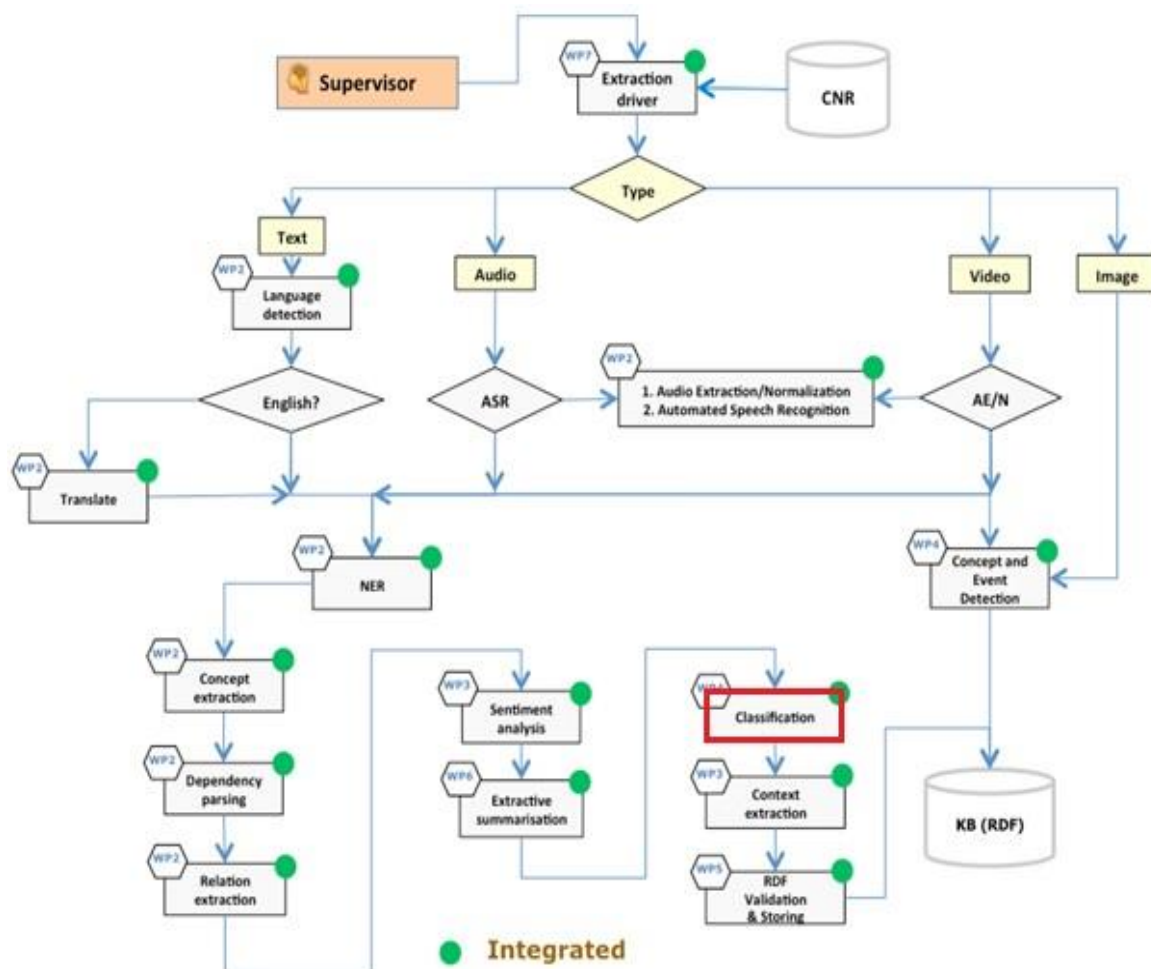


Figure 2: Content Extraction Pipeline architecture (Second Prototype) with the topic-based classification module highlighted

In Figure 3, the online modality of the Second Prototype for the MULTISENSOR platform can be seen, along with the highlighted topic-event detection, multimodal indexing and retrieval and similarity search modules. The online modality is the live connection between the user interfaces functionalities and the access to the knowledge available in the different repositories of the MULTISENSOR platform (see Section 2.4 – D7.6). It should be noted that the topic-event detection (named “Clustering” in Figure 3) and multimodal indexing and retrieval (named “Indexing” in Figure 3) modules are grouped into one block in Figure 3 just for visualization purposes.

The multimodal indexing and retrieval module stores the News Items of CNR into MongoDB, allowing for the efficient retrieval of SIMMO objects by means of the integrated multimedia retrieval framework. The topic-event detection module receives as input a list of News Items retrieved from the MongoDB, exploits multimodal features extracted from the News Items and provides a grouping of the list into a number of detected topics. Finally, the multimedia retrieval framework supports the functionality of the similarity search module, which deals with the retrieval of similar news articles based on a specific query, by utilising similarity measures from a single or multiple modalities.

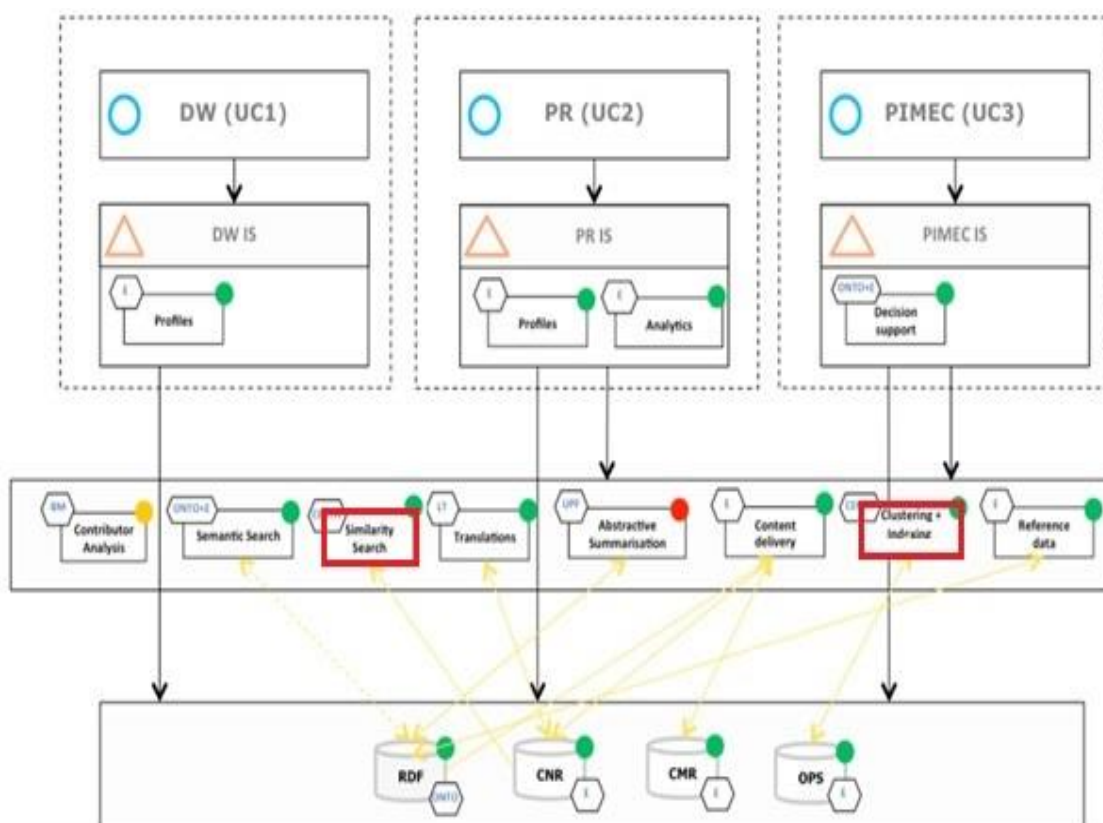


Figure 3: Online modality architecture (Second Prototype) with the topic-event detection, multimodal indexing and retrieval and similarity search modules highlighted

2 MULTIMODAL INDEXING AND RETRIEVAL

Multimedia retrieval systems become more and more popular as we need efficient and quick access to very large and diverse multimedia collections, such as video collections (eg. Youtube, Netflix) or annotated image collections (eg. Facebook, Flickr). Searching in multimedia collections becomes very challenging, due to the heterogeneous sources of information, usually textual and visual, which are combined, formulating multimedia objects: a task known as *multimodal fusion*.

Multimodal fusion (Atrey et al., 2010) merges more than one medium (referred also as modality) into one unique source of information, in order to support tasks such as multimedia search, retrieval, summarization, recommendation, clustering and classification. The modalities can be, for example, low-level visual descriptors (based on color, shape, texture, location, etc.), low-level textual features (raw text from webpages, video subtitles, or extracted from audio using automatic speech recognition, and from video using optical character recognition, etc.), metadata (time stamp, tags, source, position in a social graph) and high-level textual features (Atrey et al., 2010).

In MULTISENSOR, an efficient indexing technique for multimodal objects is developed (Tsikrika et al., 2015), as Socially Interconnected MultiMedia-enriched Objects (SIMMO). The key problem is how to combine low-level and high-level textual and visual information, in order to retrieve multimodal documents relevant to a given multimodal query. Towards this direction, we provide a novel framework for multimodal fusion of visual and textual similarities, which are based on visual features, visual concepts and textual concepts. Our method is compared to the baseline method (Ah-Pine et al., 2015), which works only for two modalities but integrates all early, late, linearly weighted, diffusion and graph-based models in one unifying framework. Moreover, the initial filtering step of the top- l text-based related documents with respect to the query allows for comparing bi-modal with multi-modal retrieval under the same memory cost.

The MULTISENSOR multimedia retrieval framework:

- fuses multiple modalities, so as to retrieve multimedia objects in response to a multimodal query;
- does not require additional memory cost;
- integrates high-level information, i.e. multimedia objects are enriched with high-level textual and visual concepts;
- is language-independent, because the textual and visual concepts are given in the same language, as provided by CEP.

Section 2.1 presents the state-of-the-art techniques in multimodal fusion and multimedia retrieval. Section 2.2 describes the necessary background and the techniques followed in the MULTISENSOR multimedia retrieval framework. In Section 2.3 it is presented the module, which implements the MULTISENSOR multimedia retrieval framework. Finally, Section 2.4 evaluates the multimedia retrieval techniques in terms of Mean Average Precision (MAP) and in terms of Average Precision (AP) per query.

2.1 Relevant work on content-based multimedia information retrieval

The multimedia retrieval problem has mainly appeared in video search engines. In (Safadi et al., 2014) a video retrieval framework is proposed: Lucene¹ text indexing on the video subtitles provides a text-based similarity score, visual concepts provide visual-based similarity scores and they are fused in a simple non-linear way (Safadi et al., 2014). Video search systems are often interactive, i.e. they fuse text-to-video and video-to-video search, where the video similarity is improved by user-generated relevance feedback and the query is progressively refined by the user (Xu et al., 2015). Otherwise, the returned results of the multimedia retrieval module are re-ranked in a post-process of the core search (Mei et al., 2014). In contradiction, the MULTISENSOR framework does not require user-generated feedback, but high-level textual and visual features, beyond visual descriptors.

Other multimedia and cross-modal retrieval tasks (Jeon et al., 2003) are motivated by Latent Dirichlet Allocation (LDA) probabilistic approaches, such as (Wang et al., 2014), which either generate a joint topic probability distribution for all modalities (Blei and Jordan, 2003), or combine the topic distribution per modality (Costa Pereira et al., 2014). Each query is related to a topic and the retrieved documents are assigned a topic distribution. If the topic distribution of a retrieved document is maximized at the query's topic, the document is relevant. Convolutional Neural Networks (CNN) have been used to learn high-level features and combine two modalities (text-image pairs) (Wang et al., 2015) for cross-modal retrieval. Another approach that aims at efficient cross-modal retrieval is by using correlation matching (Rasiwasia et al., 2010) between the two modalities. A Partial Least Squares (PLS) based approach is used in (Siddiquie et al., 2014), in order to map different modalities of the data into a common latent space and evaluate the method in the image retrieval task. Contrary to the abovementioned LDA-based, CNN-based, PLS-based and Correlation Matching approaches for cross-modal retrieval, our approach does not use any training stage, but proposes an unsupervised fusion of all features. In addition, the query in multimedia retrieval is a multimodal object, thus all modalities need to be exploited.

With respect to multimodal fusion approaches, these usually combine textual and visual features for retrieval purposes (Atrey et al., 2010). Metric fusion (Wang et al., 2013) is a random walk approach, which was designed to fuse different "views" of the same modality, such as SIFT, GIST and LBP visual features and has been evaluated in the image retrieval task. However, in MULTISENSOR we merge visual and textual information in order to perform multimedia retrieval.

Graph-based methods and random-walk approaches have been used in a unifying framework (Ah-Pine et al., 2015) for fusing visual and textual information in Content-Based Multimedia Information Retrieval (CBMIR). The random-walk approach for multimodal fusion originated in (Hsu et al., 2007), where the video search results are improved by fusing textual and visual information. The model (Ah-Pine et al., 2015) does not require user's relevance feedback, it is unsupervised, it includes as special cases all well-known early, late, linearly weighted, diffusion and graph-based models in one unifying framework and, finally, it is evaluated in the multimedia retrieval task. This unsupervised unifying framework is

¹ <https://lucene.apache.org/core/>

further elaborated, in MULTISENSOR, by utilising semantic information in each modality and by extending the method to multiple modalities.

2.2 The MULTISENSOR multimedia retrieval framework

In this Section, the necessary background for fusing two modalities is initially discussed and, then, the multimedia retrieval model of MULTISENSOR is presented for multiple modalities.

2.2.1 Preliminaries

In (Ah-Pine et al., 2015) it is assumed that “the text query is the main semantic source with regard to the user information”, so text-based relevance scores are initially computed, in order to filter out any element of the multimedia repository that does not belong to the top- l list, given by the pure textual similarities. This step allows for tuning the memory and the computational cost of the following multimedia retrieval framework.

After the selection of the top- l selected multimedia elements, with respect to the query q , the $l \times l$ similarity matrices S_t (textual similarity matrix) and S_v (visual similarity matrix) are computed and normalized. The proposed normalization (Ah-Pine et al., 2015) is:

$$s(d, d') \rightarrow \frac{s(d, d') - \min s(d, \cdot)}{\max s(d, \cdot) - \min s(d, \cdot)} \quad (2.1)$$

between any two documents d and d' .

The notation $s_t(q, \cdot)$ and $s_v(q, \cdot)$ is followed for the query-based similarity vectors on the textual and visual modality, respectively. The vectors $s_t(q, \cdot)$ and $s_v(q, \cdot)$ are normalized so that their elements sum to one. The dot product is denoted by “ \cdot ” and the (i, j) element of a matrix A is denoted by $A[i, j]$.

The baseline approach sets $x_{(0)} = s_t(q, \cdot)$ and $y_{(0)} = s_v(q, \cdot)$, and defines the following update rule:

$$x_{(i)} \propto \mathbf{K}(x_{(i-1)}, k) \cdot [(1 - \gamma)D \cdot (\beta S_t + (1 - \beta)S_v) + \gamma e \cdot s_t(q, \cdot)] \quad (2.2)$$

$$y_{(i)} \propto \mathbf{K}(y_{(i-1)}, k) \cdot [(1 - \gamma)D \cdot (\beta S_v + (1 - \beta)S_t) + \gamma e \cdot s_v(q, \cdot)] \quad (2.3)$$

where D is the row-normalizing matrix so that, for example, $D \cdot (\beta S_t + (1 - \beta)S_v)$ is row-stochastic, e is the $l \times 1$ vector of ones. The operator $\mathbf{K}(x, k)$ takes as input a vector x and assigns a zero value to elements whose score is strictly lower than the k -th highest score in x . For i iterations, the final ranking with respect to the query q is given by the linear combination of $s_t, s_v, x_{(i)}$ and $y_{(i)}$:

$$\text{score}(q) = a_t s_t(q, \cdot) + a_v s_v(q, \cdot) + a_{tv} x_{(i)} + a_{vt} y_{(i)} \quad (2.4)$$

under the restriction: $a_t + a_v + a_{tv} + a_{vt} = 1$.

Using the default parameters ($\beta = 0$, $\gamma = 0.3$, $i = 1$, $k = 10$, $l = 1000$), the model of Eq. (2.2) and (2.3) is simplified:

$$x_{(1)} \propto \mathbf{K}(s_t(q, \cdot), 10) \cdot [0.7D \cdot S_v + 0.3e \cdot s_t(q, \cdot)]$$

$$y_{(1)} \propto \mathbf{K}(s_v(q, \cdot), 10) \cdot [0.7D \cdot S_t + 0.3e \cdot s_v(q, \cdot)]$$

on the top-1000 results, which are returned by text-based search. The weights that are used in the linear combination of $s_t, s_v, x_{(1)}$ and $y_{(1)}$ are tuned in $\{0.1, 0.2, \dots, 0.9\}$ and the best values are compared to the uniform weighting strategy ($a_t = a_v = a_{tv} = a_{vt} = 0.25$). In the experiments of (Ah-Pine et al., 2015) we observed an incremental increase in Mean Average Precision (MAP) in one dataset and no increase in the other datasets, when the best weights are compared to the uniform weights.

The computation of S_t and S_v involves memory complexity $\mathcal{O}(l^2)$ and the complexity for the computation of $x_{(1)}$ and $y_{(1)}$ is $\mathcal{O}(kl)$. The top- l filtering step on the text domain reduces significantly the memory complexity of the overall multimedia retrieval framework.

2.2.2 Multimedia retrieval using multiple modalities

In general, more than two modalities are involved in the proposed late fusion scheme. The MULTISENSOR multimedia retrieval framework leverages 3 modalities from every multimedia object, as shown in Figure 4, namely visual features, visual concepts and textual concepts. Each modality provides a vector representation of the multimedia object through its corresponding features. The list of the top- l documents is obtained from the open-source Apache Lucene indexing tool. After the top- l filtering stage, the retrieval module computes:

- The $l \times l$ similarity matrices:
 - S_1 : similarity matrix on visual descriptors
 - S_2 : similarity matrix on visual concepts
 - S_3 : similarity matrix on textual concepts
- The $1 \times l$ query-based similarity vectors:
 - $s_1(q, \cdot)$: query-based similarity vector on visual descriptors
 - $s_2(q, \cdot)$: query-based similarity vector on visual concepts
 - $s_3(q, \cdot)$: query-based similarity vector on textual concepts

The fusion of the $l \times l$ similarity matrices $S_m, m = 1, 2, 3$ and the query-based similarity vectors $s_m(q, \cdot), m = 1, 2, 3$ is done using the W-MCSM model or the U-MCSM model, which are presented below. Two alternative ways to define a “contextual” similarity matrix are examined, weighted or uniform, given M modalities with corresponding query-based similarity vectors $s_m(q, \cdot), m = 1, 2, \dots, M$ and $l \times l$ similarity matrices $S_m, m = 1, 2, \dots, M$.

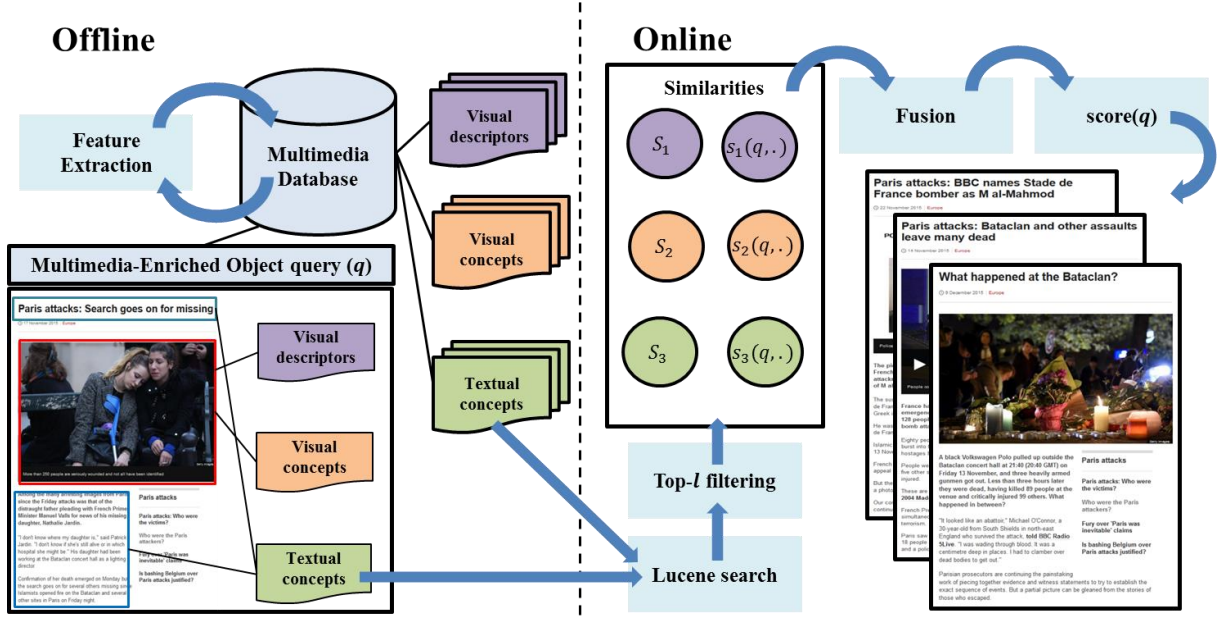


Figure 4: The MULTISENSOR multimedia retrieval framework

Weighted Multimodal Contextual Similarity Matrix (W-MCSM)

The following multimodal contextual similarity matrix is computed for each modality:

$$C_m = \left(1 - \sum_{w=1}^{M-1} \beta_w\right) S_m + \sum_{w=1}^{M-1} \beta_w S_{w \neq m}, \quad m = 1, 2, \dots, M \quad (2.5)$$

The matrices C_m , $m = 1, 2, \dots, M$ are row-normalized so as to obtain the corresponding row-stochastic transition probability matrices P_m , with elements:

$$P_m[i, j] = \frac{C_m[i, j]}{\sum_{j=1}^l C_m[i, j]} \quad (2.6)$$

For all m modalities we set: $x_{(0)}^m = s_m(q, \cdot)$, $m = 1, 2, \dots, M$, so as to define the following update rule, inspired by the model of Eq. (2.2) and (2.3):

$$x_{(i)}^m \propto K(x_{(i-1)}^m, k) \cdot \left[\left(1 - \sum_{w \neq m} \gamma_w\right) P_m + \sum_{w \neq m} \gamma_w S_w(q, \cdot) \right] \quad (2.7)$$

Motivated by Eq. (2.4), the vector of relevance score in response to the query q is:

$$score(q) = \sum_{m=1}^M a_m s_m(q, \cdot) + \sum_{m=1}^M a'_m x_{(i)}^m \quad (2.8)$$

under the restriction that:

$$\sum_{m=1}^M a_m + \sum_{m=1}^M a'_m = 1 \quad (2.9)$$

For example, in the case of 3 modalities, the model W-MCSM is illustrated as follows. Formula (2.5) reduces to:

$$C_1 = (1 - \beta_1 - \beta_2) S_1 + \beta_1 S_2 + \beta_2 S_3$$

$$C_2 = (1 - \beta_1 - \beta_2)S_2 + \beta_1S_1 + \beta_2S_3$$

$$C_3 = (1 - \beta_1 - \beta_2)S_3 + \beta_1S_1 + \beta_2S_2$$

The multimodal contextual similarity matrices $C_m, m = 1, 2, 3$ are row-normalized to obtain $P_m, m = 1, 2, 3$, using Eq. (2.6). The update rule (2.7) becomes:

$$x_{(i)}^1 \propto \mathbf{K}(x_{(i-1)}^1, k) \cdot [(1 - \gamma_2 - \gamma_3)P_1 + \gamma_2e \cdot s_2(q, \cdot) + \gamma_3e \cdot s_3(q, \cdot)]$$

$$x_{(i)}^2 \propto \mathbf{K}(x_{(i-1)}^2, k) \cdot [(1 - \gamma_1 - \gamma_3)P_2 + \gamma_1e \cdot s_1(q, \cdot) + \gamma_3e \cdot s_3(q, \cdot)]$$

$$x_{(i)}^3 \propto \mathbf{K}(x_{(i-1)}^3, k) \cdot [(1 - \gamma_2 - \gamma_1)P_3 + \gamma_2e \cdot s_2(q, \cdot) + \gamma_1e \cdot s_1(q, \cdot)]$$

The vector of relevance $score(q)$, in response to the query q , is computed as in Eq. (2.8) and linearly combines $s_m(q, \cdot), m = 1, 2, 3$ and $x_{(i)}^m, m = 1, 2, 3$:

$$score(q) = a_1s_1(q, \cdot) + a_2s_2(q, \cdot) + a_3s_3(q, \cdot) + a'_1x_{(i)}^1 + a'_2x_{(i)}^2 + a'_3x_{(i)}^3$$

under the restriction $a_1 + a_2 + a_3 + a'_1 + a'_2 + a'_3 = 1$.

A variation of the W-MCSM model is the U-MCSM, which uses one (uniform) multimodal contextual similarity matrix for all modalities.

Uniform Multimodal Contextual Similarity Matrix (U-MCSM)

The following multimodal contextual similarity matrix is computed for all modalities:

$$C = \sum_{m=1}^M \beta_m S_m, \quad \sum_{m=1}^M \beta_m = 1 \quad (2.10)$$

The matrix C of Eq. (2.10) is row-normalized so as to obtain the row-stochastic matrix P :

$$P[i, j] = \frac{C[i, j]}{\sum_{j=1}^l C[i, j]} \quad (2.11)$$

For all m modalities we set $x_{(0)}^m = s_m(q, \cdot), m = 1, 2, \dots, M$, and we define the update rule:

$$x_{(i)}^m \propto \mathbf{K}(x_{(i-1)}^m, k) \cdot \left[\left(1 - \sum_{w \neq m} \gamma_w \right) P + \sum_{w \neq m} \gamma_w s_w(q, \cdot) \right] \quad (2.12)$$

The vector of relevance score in response to the query q is given by Eq. (2.8), under the restriction of Eq. (2.9), as also in W-MCSM.

Memory Complexity

The memory complexity is $\mathcal{O}(l^2)$ for the computation of each similarity matrix $S_m, m = 1, 2, \dots, M$, $\mathcal{O}(l)$ for each similarity vector $s_m(q, \cdot)$ and $\mathcal{O}(kl)$ for each $x_{(i)}^m$, thus the overall memory complexity is quadratic in l : $\mathcal{O}(Ml^2 + Mkl + Ml)$.

In order to compare directly the use of two modalities with the MULTISENSOR retrieval framework with $M > 2$ modalities, under the same memory complexity, we seek for the number of filtered documents l' , such that:

$$Ml'^2 + Mkl' + Ml' = 2l^2 + 2kl + 2l \quad (2.13)$$

The non-negative solution of Equation (2.13) with respect to l' is:

$$l' = \sqrt{\frac{(k+1)^2}{4} + \frac{2l^2 + 2kl + 2l}{M}} - \frac{k+1}{2} \quad (2.14)$$

Modalities	l	Modalities	l
2	1000	9	468
3	815	10	444
4	704	11	423
5	630	12	405
6	575	13	389
7	531	14	374
8	497	15	361

Table 1: The proposed values l for the top- l filtering step

Table 1 reports the l' values (for the defaults with 2 modalities: $k = 10, l = 1000$), so as to avoid significant memory increase when multiple modalities are fused. For example, in the case of $M = 3$: $l' \cong 815$ and even for 15 modalities, the number of the top- l filtered documents remains higher than 300, hence a critical number of documents are involved in the multimodal fusion. The non-linear decrease in l' , as the number of modalities increase, is shown in Figure 5.

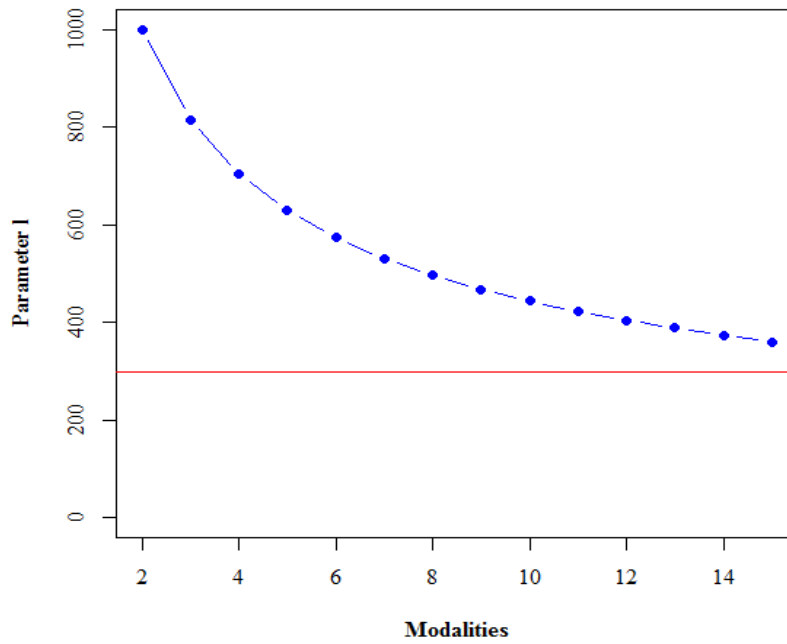


Figure 5: The selection of the parameter l

2.3 Multimodal indexing and retrieval module

2.3.1 Multimodal indexing

In previous deliverable D4.1, we have developed a multimedia indexing model, namely SIMMO, which is implemented on Github², and provides efficient representation of a multimedia object in any document oriented database, such as MongoDB³. However, we slightly updated the model in terms of speed, when multiple modalities of multiple queries are required. We added the webpage IDs in each class (modality), as shown in Figure 6, in order to reduce communication from one class to another, in the case of cross-modal queries in the database.

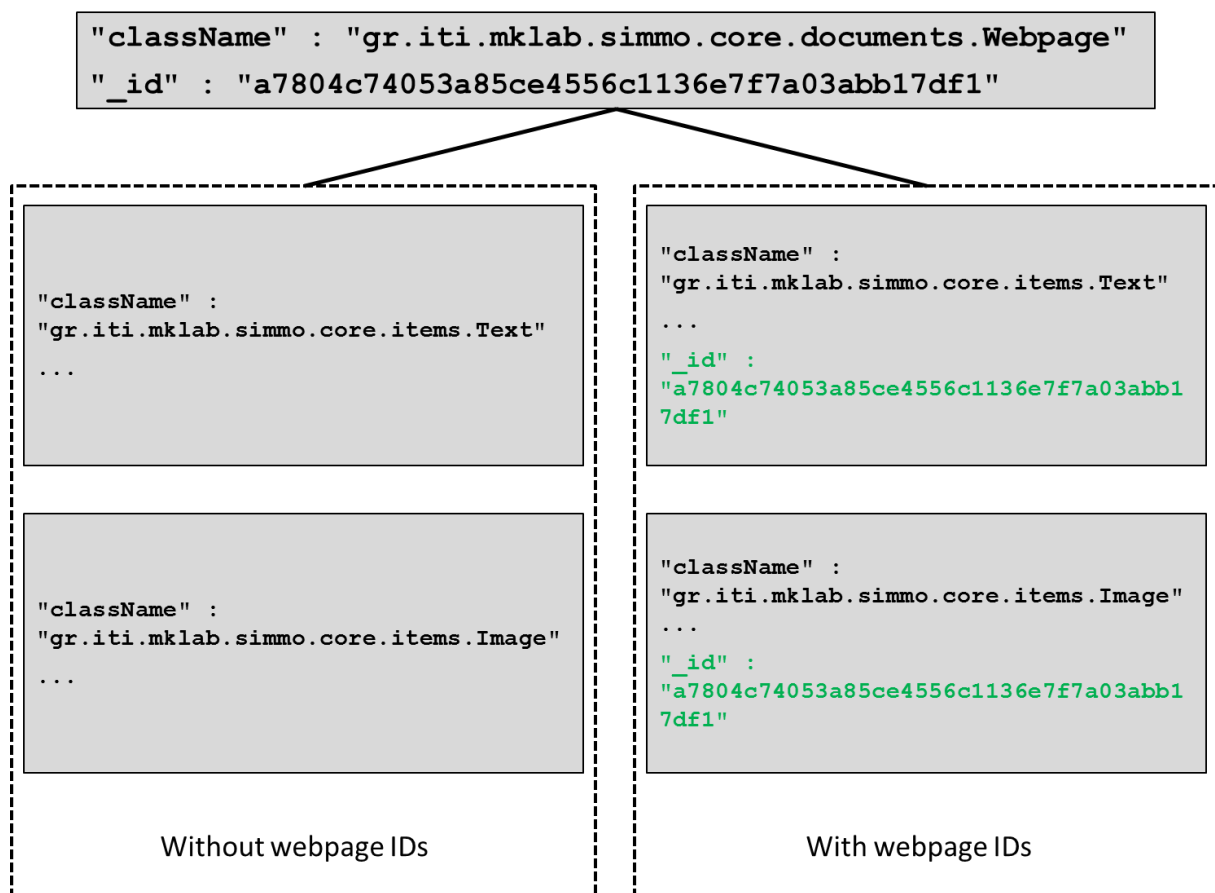


Figure 6: Direct linking of textual and visual modalities with webpage IDs

² <https://github.com/MKLab-ITI/simmo>

³ <https://www.mongodb.org/>

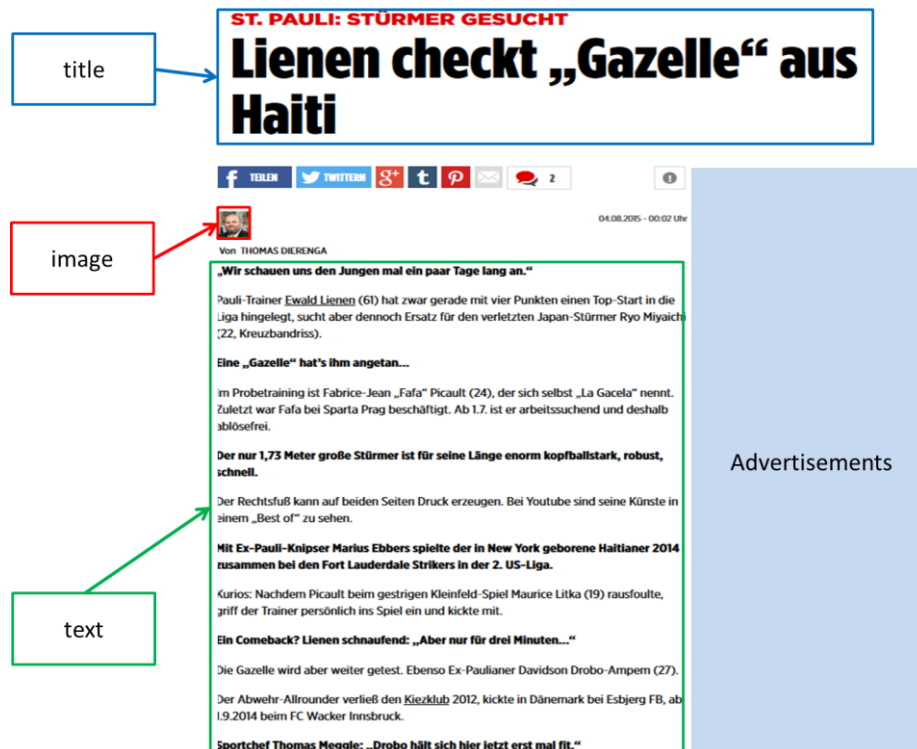


Figure 7: A webpage from a German sports article

Consider, for instance, a Web page from a German sports article (Figure 7). SIMMO models this as a Webpage corresponding to the following JSON:

```
<!-- Webpage.json -->
{
  "_id" : "a7804c74053a85ce4556c1136e7f7a03abb17df1",
  "className" : "gr.iti.mklab.simmo.core.documents.Webpage",
  "source" : "Bild Zeitung",
  "items" : [
    {
      "$ref" : "Image",
      "$id" : "780ce47d-9f17-4b95-9008-84027eb8a127"
    },
    {
      "$ref" : "Image",
      "$id" : "a655ccec-4805-4c5e-9549-debd93ad0bf3"
    },
    ...
    {
      "$ref" : "Text",
      "$id" : "6df6b329-9036-4b6a-a8ff-2505e462681e"
    }
  ],
  "language" : "de",
  "country" : "DE",
  "url" : "http://www.bild.de/sport/fussball/st-pauli/lienen-checkt-gazelle-
aus-haiti-42050050.bild.html",
  "title" : "Ewald Lienen Pauli-Trainer checkt „Gazelle“ aus Haiti",
  "creationDate" : ISODate("2015-08-03T23:29:59.000Z"),
  "annotations" : [
    {
      "className" :
"gr.iti.mklab.simmo.core.annotations.SentimentPolarity",
      "positive" : 0,
      "negative" : 0,
      "sentiment" : 0,
```

```

        "polarity" : 0
    },
    {
        "className" : "gr.iti.mklab.simmo.core.annotations.Category",
        "category" : "Headlines,2. Bundesliga,Dierenga Thomas,Ablöse,FC St.
Pauli,Lienen Ewald"
    }
]
}

```

The corresponding textual features of the Webpage (Figure 7) are:

```

<!-- Text.json -->
{
    "_id" : "6df6b329-9036-4b6a-a8ff-2505e462681e",
    "className" : "gr.iti.mklab.simmo.core.items.Text",
    "txtType" : "TXT",
    "content" : ": sought striker Ewald Lienen ... ",
    "type" : "TEXT",
    "annotations" : [
        {
            "className" : "gr.iti.mklab.simmo.core.annotations.Concepts",
            "_id" : "564c33b27819cf07b8164807",
            "conceptsList" : [
                {
                    "_id" : "564c33b27819cf07b81647df",
                    "conceptModality" : "TEXTUAL",
                    "concept" : "Perception_active",
                    "score" : 1
                },
                {
                    "_id" : "564c33b27819cf07b81647e0",
                    "conceptModality" : "TEXTUAL",
                    "concept" : "Closure",
                    "score" : 1
                },
                ...
                {
                    "_id" : "564c33b27819cf07b8164806",
                    "conceptModality" : "TEXTUAL",
                    "concept" : "Elaboration.Change_of_leadership",
                    "score" : 1
                }
            ]
        },
        {
            "className" : "gr.iti.mklab.simmo.core.annotations.NamedEntities",
            "_id" : "564c33b27819cf07b8164828",
            "namedEntitiesList" : [
                {
                    "_id" : "564c33b27819cf07b8164808",
                    "namedEntityType" : "LOCATION",
                    "token" : "Prague",
                    "count" : 0
                },
                {
                    "_id" : "564c33b27819cf07b8164809",
                    "namedEntityType" : "LOCATION",
                    "token" : "Japan",
                    "count" : 0
                },
                ...
                {
                    "_id" : "564c33b27819cf07b8164827",
                    "namedEntityType" : "TIME",
                    "token" : "2015",

```

```

        "count" : 0
      }
    ]
  }
}

```

An image of the Webpage of Figure 7 is represented as:

```

<!-- Image.json -->
{
  "_id" : "a655ccec-4805-4c5e-9549-debd93ad0bf3",
  "className" : "gr.iti.mklab.simmo.core.items.Image",
  "size" : NumberLong(0),
  "width" : 291,
  "height" : 164,
  "numLikes" : 0,
  "numShares" : 0,
  "numComments" : 0,
  "numViews" : 0,
  "numRatings" : 0,
  "type" : "IMAGE",
  "url" : "http://bilder.bild.de/fotos-skaliert/bvb-heiss-auf-den-schweizer-
dortmunds-shaqiri-plan-46924812-42039186/3,w=291,c=0.bild.jpg",
  "annotations" : [
    {
      "className" :
"gr.iti.mklab.simmo.core.annotations.lowleveldescriptors.LocalDescriptors",
      "descriptorType" : "SIFT",
      "featureEncodingType" : "Vlad",
      "numberOfFeatures" : 0,
      "featureEncodingValue" : "-0.0153925 0.0306615 0.0295825 ... "
    },
    {
      "className" :
"gr.iti.mklab.simmo.core.annotations.lowleveldescriptors.LocalDescriptors",
      "descriptorType" : "SIFT_OPPONENT",
      "featureEncodingType" : "Vlad",
      "numberOfFeatures" : 0,
      "featureEncodingValue" : "-0.0446101 -0.00594323 0.00838053 ... "
    },
    {
      "className" :
"gr.iti.mklab.simmo.core.annotations.lowleveldescriptors.LocalDescriptors",
      "descriptorType" : "SURF",
      "featureEncodingType" : "Vlad",
      "numberOfFeatures" : 0,
      "featureEncodingValue" : "-0.0332221 0.0304886 0.0986858 ... "
    },
    {
      "className" :
"gr.iti.mklab.simmo.core.annotations.lowleveldescriptors.LocalDescriptors",
      "descriptorType" : "SURF_OPPONENT",
      "featureEncodingType" : "Vlad",
      "numberOfFeatures" : 0,
      "featureEncodingValue" : "-0.0106512 0.0428385 0.0247903 ... "
    },
    {
      "className" :
"gr.iti.mklab.simmo.core.annotations.lowleveldescriptors.LocalDescriptors",
      "descriptorType" : "SURF_RGB",
      "featureEncodingType" : "Vlad",
      "numberOfFeatures" : 0,
      "featureEncodingValue" : "-0.0293861 0.0440064 0.0880238 ... "
    }
  ]
}

```

```
"className" : "gr.itl.mklab.simmo.core.annotations.Concepts",
"_id" : "564d932c7819cf1b48baa89c",
"conceptsList" : [
  {
    "_id" : "564d932c7819cf1b48baa882",
    "conceptModality" : "VISUAL",
    "concept" : "Body_Parts",
    "score" : 0.884228
  },
  {
    "_id" : "564d932c7819cf1b48baa883",
    "conceptModality" : "VISUAL",
    "concept" : "Eukaryotic_Organism",
    "score" : 0.810715
  },
  ...
  {
    "_id" : "564d932c7819cf1b48baa887",
    "conceptModality" : "VISUAL",
    "concept" : "Man_Made_Thing",
    "score" : 0.739466
  }
]
}
```

From each SIMMO object, we get its textual concepts, the visual descriptors and the visual concepts of the most representative image. The features, which are employed in the MULTISENSOR multimedia retrieval framework, are listed as follows:

Visual descriptors: The scale-invariant local descriptors RGB-SIFT (Van De Sande et al., 2010) are extracted and, then they are locally aggregated into one vector (4000-dimensional) representation using VLAD encoding (Jégou et al., 2010).

Visual concepts: The images of the multimedia objects are indexed by 346 high-level concepts (e.g. water, aircraft), which are detected by multiple independent concept detectors. The locally aggregated features (VLAD encoding for RGB-SIFT descriptors) are served as input to Logistic Regression classifiers and their output is averaged and further refined (Safadi and Quénot, 2011).

Textual concepts (TC): The textual concepts used in evaluation of the retrieval module (Section 2.4) are extracted using the DBpedia Spotlight⁴ annotation tool, which is an open source project for automatic annotation of DBpedia entities in natural language text (Daiber et al., 2013).

For the textual concepts, Lucene indexing provides the similarity score between any two text documents, where Lucene-based similarity⁵ scores are obtained by “Lucene’s Practical Scoring Function”. The visual modalities (descriptors, concepts) are provided as vector representations for each multimedia object, so for the computation of the similarity between any two objects we follow (Hafner et al., 1995). In brief, let D_{ij} be the Euclidian

⁴ <http://dbpedia-spotlight.github.io/demo/>

⁵ https://lucene.apache.org/core/3_0_3/api/core/org/apache/lucene/search/Similarity.html

distances for all pairs (i, j) of objects and let also $d_{max} = \max D_{ij}$ be the maximum of all D_{ij} , then the similarity S_{ij} between object i and object j is:

$$S_{ij} = 1 - \frac{D_{ij}}{d_{max}} \quad (2.15)$$

The similarities of Equation (2.15) take values in the interval $[0, 1]$.

2.3.2 Multimedia retrieval module

The module constructs the similarity matrices of the multimedia retrieval framework (Figure 4) and fuses them for the computation of one relevance score vector, which is uniform for all modalities: $score(q)$. The variation U-MCSM of the model W-MCSM does not provide statistically different MAP scores, as shown later in the evaluation results, thus U-MCSM and W-MCSM outperform the baseline SoA method (Ah-Pine et al., 2015) in equivalent ways. MULTISENSOR uses W-MCSM for multimodal fusion of all similarities and the overall multimedia retrieval framework, after the top- l filtering step is implemented in Python⁶. The code developed within MULTISENSOR with respect to the multimedia retrieval module is available at: <https://github.com/MKLab-ITI/multimedia-retrieval>.

2.4 Application to MULTISENSOR Use Cases and evaluation

In the following, we describe the datasets that are used for evaluation of the multimedia retrieval framework and the corresponding results. The datasets are chosen so that they contain multimodal queries and objects, where textual metadata are available for all objects.

2.4.1 Evaluation data

The MULTISENSOR multimedia retrieval framework is evaluated in two datasets, namely the IAPR-TC12⁷ dataset and the WIKI11⁸ dataset of sizes 20,000 and 237,434 respectively. The 20,000 images of IAPR-TC12 include pictures of sports, actions, people, animals, cities, landscapes and many other topics. The IAPR-TC12 and the WIKI11 datasets have been annotated by means of the ImageCLEF campaign (Grubinger et al., 2006; Tsirikas and Kludas, 2010). The title and the description of each image are utilized in order to form multimodal objects, as shown in Figure 8.

The WIKI11 dataset has 237,434 images with description in one to three languages and 50 topics with one to five query images with caption. The IAPR-TC12 dataset has 60 queries with 3 examples per query.

⁶ <https://www.python.org/>

⁷ <http://imageclef.org/photodata>

⁸ <http://www.imageclef.org/wikidata>

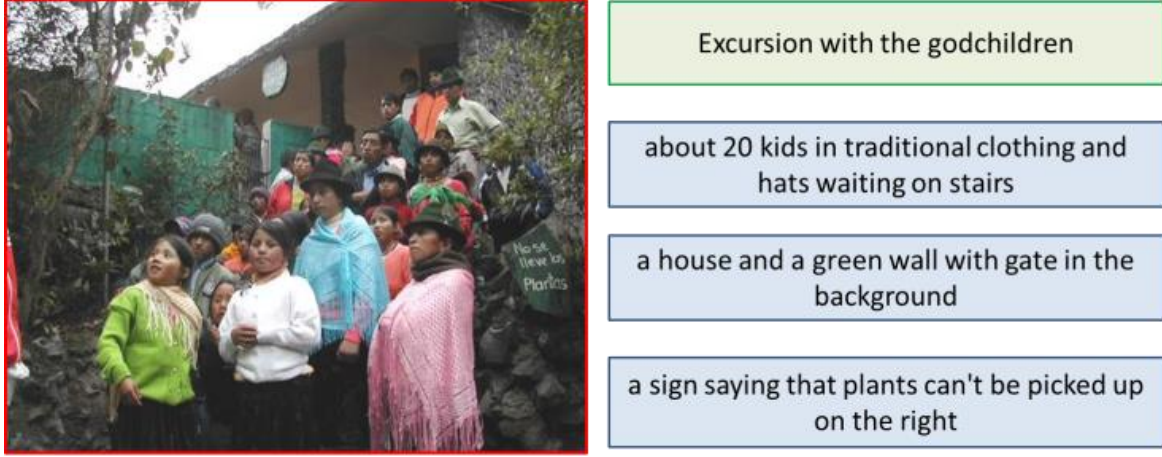


Figure 8: An annotated image with title and description

2.4.2 Evaluation results

The performance of the MULTISENSOR multimedia retrieval framework is evaluated using the Mean Average Precision, which is broadly used for Information Retrieval tasks. Given a query, the average precision is computed, defined as the area under the precision-recall curve and, then, averaging for all queries we obtain the MAP score for each dataset.

In order to compare directly the baseline SoA method (Ah-Pine et al., 2015) with the MULTISENSOR framework with three modalities, under the same memory complexity, we select the number of filtered documents l , according to Table 1, such that the overall memory complexity of the baseline coincides with the overall memory complexity of the proposed models U-MCSM and W-MCSM, thus we get $l = 815$.

Parameters			WIKI11		IAPR-TC12	
Baseline (1 modality - best)			0.2611		0.2314	
Baseline (2 modalities)			0.3654		0.2769	
γ_1	γ_2	γ_3	W-MCSM	U-MCSM	W-MCSM	U-MCSM
0.00	0.00	1.00	0.3637	0.3637	0.3065	0.3065
0.00	1.00	0.00	0.3432	0.3433	0.2858	0.2857
1.00	0.00	0.00	0.3854	0.3855	0.2518	0.2518
0.50	0.50	0.00	0.4086	0.4083	0.2912	0.2912
0.5	0.25	0.25	0.4146*	0.4145*	0.2969	0.2970
0.25	0.50	0.25	0.4029	0.4029	0.3136	0.3136
0.25	0.25	0.50	0.4048	0.4048	0.3204*	0.3204*
0.34	0.33	0.33	0.4104	0.4105	0.3148	0.3148

Table 2: Mean Average Precision for the WIKI11 and the IAPR-TC12 datasets. We mark in bold the MAP values which outperform the baseline and with “*” the best MAP

A direct comparison between the two methods U-MCSM and W-MCSM, in terms of Mean Average Precision, shows that MAP scores differ less than 1.00% in all cases examined. They both outperform the baseline in approximately equivalent ways.

In Table 2, we observe that the best MAP scores appear when $\gamma_1 = 0.5, \gamma_2 = 0.25, \gamma_3 = 0.25$ for the WIKI11 dataset and when $\gamma_1 = 0.25, \gamma_2 = 0.25, \gamma_3 = 0.5$ for the IAPR-TC12 dataset. For the best values of $\gamma_i, i = 1, 2, 3$ we then tuned the parameters $\beta_i, i = 1, 2, 3$ under the assumption $\beta_1 + \beta_2 + \beta_3 = 1$ and we did not observe any triplet $(\beta_1, \beta_2, \beta_3)$ that significantly improves Mean Average Precision either in W-MCSM or in U-MCSM. For the formulation of the relevance score vector $score(q)$, we adopt a uniform weighting strategy $a_1 = a_2 = a_3 = a'_1 = a'_2 = a'_3 = \frac{1}{6}$ and tune the values $a_i, a'_i, i = 1, 2, 3$ with step 0.25 under the assumption $a_1 + a_2 + a_3 + a'_1 + a'_2 + a'_3 = 1$, when $\beta_i, \gamma_i, i = 1, 2, 3$ are kept fixed with their best values, no further increase in MAP is observed.

Finally, we observe that the special case of equal $\gamma_i, i = 1, 2, 3$ shows little variation from the best $\gamma_i, i = 1, 2, 3$. In order to employ an automatic unsupervised method for multimedia retrieval, we used the values $\gamma_i = 1/3, i = 1, 2, 3$ and performed experiments on the present database of MULTISENSOR. In case an object has multiple images, we select the one which maximizes content, quantified by the number of visual concepts. We manually annotated the top-20 retrieved documents for 5 indicative queries, from which two are in German and three in English (Table 3).

The results of Table 3 show the superiority of our approach, when compared to the general and unifying baseline framework (Ah Pine et al., 2015). The second and the fourth queries show an incremental increase in MAP of 0.81% and 0.90% respectively, but the increase in MAP is 5.02% at the third query. Finally, for the fifth query, the increase in Average Precision is 2.03%. We mention that due to the absence of image in the first query, Average Precision is the same for both methods because there is only one modality in the query.

Query	Query (title)	AhPine2015	MULTISENSOR
1	IFA/ROUNDUP Springer sichert sich Platz auf Samsung-Smartphones	0.5768	0.5768
2	How to save money on energy	0.8229	0.8296
3	1000 neue Jobs! Siemens baut Windkraft- Werk in Cuxhaven	0.6996	0.7347
4	Five favourite crumble recipes for autumn	0.9342	0.9426
5	Investors brace for stocks to fall again ahead of earnings	0.7491	0.7643

Table 3: Average Precision

The time performance of W-MCSM with 3 modalities is compared to the time performance of the baseline approach with 2 modalities. In the case of 3 modalities, we observe an incremental increase in processing time. However, when Equation (2.14) is involved in the reduction of memory complexity, in order to compare the 3-modalities approach with the 2-modalities approach under the same memory complexity, W-MCSM-memo becomes 14.97%

faster in the case of $l = 500$ retrieved documents and 22.17% faster in the case of $l = 1000$ top- l filtered documents.

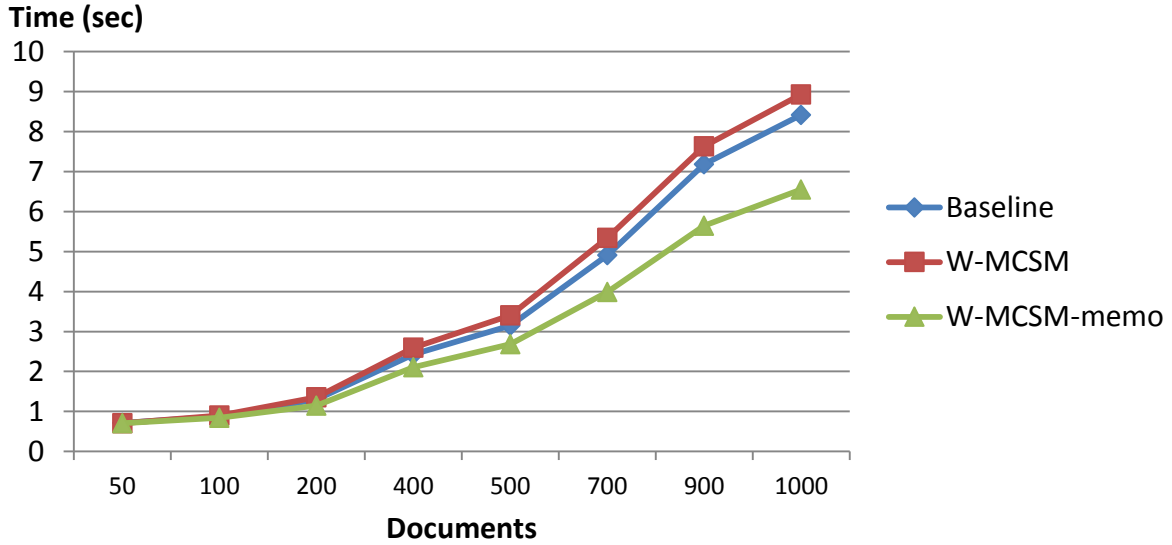


Figure 9: Time performance of the baseline method and our W-MCSM model

We presented a novel framework for multimedia retrieval which, in general, fuses M modalities. In particular, when our approach is applied to textual concepts, visual features and visual concepts ($M=3$ modalities), the MAP increases up to 15.71% for the datasets WIKI11 and IAPR-TC12. It should be noted that according to the quality metrics established in deliverable D1.2 (Self-assessment plan v2) for T4.4, the multimedia retrieval task has been achieved to the highest expectation for the datasets WIKI11 and IAPR-TC12, as there is an improvement of MAP by almost 16%, compared to the baseline system. Regarding the MULTISENSOR present database, the proposed multimedia retrieval framework is faster (Figure 9) and more efficient (Tables 2 and 3) than the baseline unifying framework.

3 TOPIC-BASED CLASSIFICATION

In deliverable D4.1, an initial multimodal framework for classifying news articles retrieved from the Central News Repository (CNR) into predefined categories, based on the Random Forests (RF) machine learning method, was proposed. In addition, this framework made use of late fusion strategies based on the operational capabilities of RF. We selected two modalities, in order to suitably represent each news article for the classification process, namely a) N-gram textual features extracted from the textual description of each article and b) low-level visual features extracted from the biggest image (assumed to be the representative one) of each article.

For the final topic-based classification framework that is presented in the current deliverable, due to the fact that a) one main finding from the experiments conducted in D4.1 was that the textual modality is more reliable and suitable for the topic-based classification task than the visual one and b) it is not guaranteed that all news articles contain one or more images, in order to be able to extract visual features from them, we decided to replace the visual modality with another textual-based modality, which exploits a recently introduced methodology called word2vec that is used for producing word embeddings. It should be noted that we decided not to address rule-based classification, as originally planned, since this would require extensive experimentation with a Knowledge Base (KB) that is populated with a very large amount of SIMMOs and the KB population task is still in progress.

The relevant work on category-based classification, with respect to the tasks of Document classification (DC), News Items classification and classification using multimodal features, has already been presented in deliverable D4.1 (see Section 4.1 – D4.1). Therefore, we will not go into any further details on this subject. Instead, we will focus on the new element of our framework, the word2vec methodology. Below, we provide a brief introduction to word2vec and some relevant work.

Word2vec

Word embeddings can be defined as functions, whose aim is to represent words from a given vocabulary as continuous vectors in a low-dimensional space (relative to the size of the vocabulary)⁹. Mikolov et al. (2013) proposed novel architectures and models for producing word embeddings, based on deep neural networks. Specifically, the Continuous Bag-of-Words (CBOW) and the Skip-gram models were introduced. In the literature, they are also referred to as word2vec. They are both similar to the Feedforward Neural Net Language Model (NNLM) and they consist of an input, a projection and an output layer. These models make use of the context of the words, given a large corpus, where context is the neighbouring words in a sentence. The context size one can take into consideration is specified by a parameter called *window*. In the CBOW architecture, the NN model tries to predict a word given the context of this word, whereas in the Skip-gram architecture, the exactly opposite function is executed, that is, given a word the NN model tries to predict the context of a word.

⁹ https://en.wikipedia.org/wiki/Word_embedding

Regarding the quality of these vectors, it is proved that these methods can capture very efficiently the semantics of the words. Words with similar meanings will have a similar context in the corpus and thus, the trained vectors will be close to each other. Furthermore, it can capture relationships between words in a way that we can answer simple questions by performing simple operations in the vectors. For example, consider the following question: “Which word is related to the word *Athens* in the same sense that word *Spain* is related to the word *Madrid*?”. It can be answered by simply calculating vector $X = \text{vector}(\text{“Spain”}) - \text{vector}(\text{“Madrid”}) + \text{vector}(\text{“Athens”})$. The model can capture the country-capital relationship between the pairs of words and the resulting vector X will be very similar to vector (‘‘Greece’’). The superiority of word2vec over simpler methods (e.g. Latent Semantic Analysis) drew the attention of many other works (see for example (Xing et al., 2014), (Ju et al., 2015), (Lilleberg et al., 2015) and (Zhang et al., 2015)), which utilized the methodology, in order to train efficient document and sentiment classification models.

3.1 The MULTISENSOR category-based classification framework based on multimodal features

In this Section, since the final category-based classification framework follows exactly the same approaches as the initial classification framework (with the exception of the visual modality, which is replaced by a modality that utilizes word2vec feature vectors) and in order to facilitate the presentation of the deliverable, we repeat all the information regarding the theoretical background of RF and the basic elements of the proposed category-based classification framework, as described in D4.1 (for more information see Section 4.2 – D4.1).

Theoretical background

Random Forests (RF) is an ensemble learning method for classification and regression (Breiman, 2001). The basic notion of the methodology is the construction of a multitude of decision trees. Within RF’s operational procedures, two sources of randomness are employed:

1. A different bootstrap sample, drawn randomly from the training data, is used for the construction of each decision tree.
2. At each node split during the construction of a decision tree, a random subset of p variables is selected from the original variable set and the best split based on these p variables is used.

For the prediction of an unknown case, the outputs of the trees that are constructed by RF are aggregated (majority voting for classification / averaging for regression). For a model consisting of T trees, the following equation is used for predicting the class label l of a case y through majority voting:

$$l(y) = \underset{c}{\operatorname{argmax}} \left(\sum_{n=1}^T I_{h_n(y)=c} \right) \quad (3.1)$$

where I denotes the indicator function and h_n the n th tree of the RF.

RF can provide an estimation of its generalization error through an internal mechanism called Out-Of-Bag (OOB) error estimate. During the construction of each tree, only 2/3 of the

original data's cases are used in that particular bootstrap sample. The rest 1/3 of the instances (OOB data) are predicted by the constructed tree and thus, used for testing its performance. The OOB error estimate is the averaged prediction error for each training case y , using only the predictions of the trees that do not contain y in their bootstrap sample. In general, it is not considered necessary to perform cross-validation during the training of a RF model. This is because the OOB error estimate is actually an indicative performance score based on cases that do not take part in the training procedure of RF (the OOB data). Furthermore, RF can supply a matrix that contains proximities between the training cases. This is achieved by putting all the training cases down each tree and based on the frequency that pairs of cases end up in the same terminal nodes, this proximity matrix is computed.

Proposed classification framework

In Figure 10, the flowchart of the proposed classification framework (training phase) is illustrated. Next, the different steps and notions of the framework are described in detail.

First of all, it is assumed that each News Item is represented by a number of modalities. By applying certain procedures, we extract a number of features from the raw data of each modality, thus formulating the corresponding feature vectors that serve as input for the construction of the classification models. At this point, it should be noted that we chose to follow the approach of treating each modality's features separately, instead of concatenating all the features into one large vector. In this way, we are able to exploit the representation and the information contained in each modality in an independent manner.

As a result of the aforementioned approach, in the training phase a separate RF model is trained for each modality. In order to formulate a final fused RF model, we apply a late fusion strategy by computing weights for each modality's RF outputs. For the computation of the modality weights, three different methods that exploit the operational procedures of RF are applied:

OOB error estimate: We assume that if a RF model is able to predict the OOB cases for one or more classes efficiently, it is expected to perform equally well on unknown cases. Therefore, from each modality's RF model, the corresponding OOB accuracy values are computed. This is done for each class separately. Then, the accuracy values are normalized (by dividing them by their sum) and serve as weights for the RF models' outputs, e.g. for class l :

- acc_{OOBli} : OOB accuracy value for class l for modality i ($i=1...N$, N =number of modalities)
- $W_{li} = \frac{acc_{OOBli}}{\sum_{j=1}^N acc_{OOBli}}$ (weight for class l for modality i) (3.2)

Proximity ratio: For the second weighting strategy, the proximity matrix of a RF model is taken into consideration. First, for each RF the proximity matrix between all pairs of data cases $P=\{p_{ij}, i,j =1, ...,w\}$ (w =number of data cases) is constructed. Next, the proximity ratio values between the inner-class and the intra-class proximities (for each class) are computed (Zhou et al., 2010) as in the following equation:

$$R = \frac{P_{inner}}{P_{intra}} \quad (3.3)$$

where

$$P_{inner} = \sum_{i,j=1}^w p_{ij} \text{ (if } l_i = l_j \text{)} \quad (3.4)$$

$$P_{intra} = \sum_{i,j=1}^w p_{ij} \text{ (if } l_i \neq l_j \text{)} \quad (3.5)$$

and l_i, l_j denote the class labels of cases i and j , respectively. Finally, for each modality and for each class, the proximity ratio values are first averaged and then normalized (by dividing them by their sum), in order to be used as modality weights for the RF models, e.g. for class l :

- meanR_{li} : Averaged proximity ratio value for class l for modality i ($i=1\dots N$, N =number of modalities)
- $W_{li} = \frac{\text{meanR}_{li}}{\sum_{j=1}^N \text{meanR}_{lj}}$ (weight for class l for modality i)

(3.6)

A large proximity ratio value for a class is an indication that the cases of that class are encountered frequently in the terminal nodes of a RF model's trees (inner-class proximity) and are not intermixed with cases from other classes (intra-class proximity). Thus, the larger the proximity ratio value for a class, the better the performance of the RF model for that class can be considered.

Adjusted proximity ratio: This approach utilizes the two aforementioned weighting strategies (OOB error estimate and proximity ratio). It is used for adjusting the proximity ratio values, in cases where one or more classes for a modality's RF model exhibit high averaged proximity ratio values but disproportionally low OOB accuracy values. As a result, the weights assigned to these classes will be biased towards the "worse" modality (in terms of accuracy performance) and this will affect the late fused RF outputs. To overcome this, for each class and for each modality, the averaged proximity ratio values are multiplied by the corresponding OOB accuracy values, in order to formulate the adjusted proximity ratio values as in the following equation:

$$R_{adjusted} = R * OOB_{accuracy} \quad (3.7)$$

After the computation of the adjusted proximity ratio values, the same normalization procedure (as in the other two weighting strategies) is applied, e.g. for class l :

- meanRad_{li} : Averaged adjusted proximity ratio value for class l for modality i ($i=1\dots N$, N =number of modalities)
- $W_{li} = \frac{\text{meanRad}_{li}}{\sum_{j=1}^N \text{meanRad}_{lj}}$ (weight for class l for modality i)

(3.8)

During the testing phase, for the prediction of an unknown case, RF outputs probability estimates per class for that case. The probability outputs P_1, P_2, \dots, P_N (N =number of modalities) from the RF models are multiplied by their corresponding modality weights W_1, W_2, \dots, W_N and summed to produce the final RF predictions, e.g. for class l :

$$P_l^{fused} = W_{l1}P_{l1} + W_{l2}P_{l2} + \dots + W_{lN}P_{lN} \quad (3.9)$$

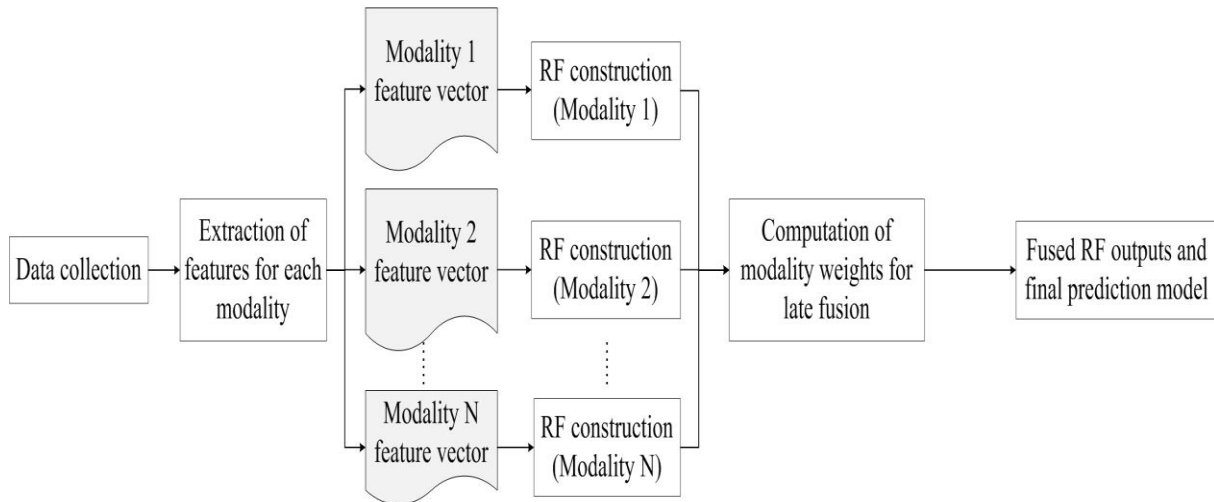


Figure 10: Proposed classification framework (training phase)

3.2 Category-based classification module

The code for the experiments conducted within MULTISENSOR for the category-based classification task has been developed in Python¹⁰, version 3.5.1, 64-bit, (the experiments are presented in Section 3.3). It makes use of many external packages such as *gensim*, *nltk*, *numpy* and *sklearn*. The dataset used in these experiments contains news documents, which are categorized into six classes (one folder per class). Two types of features are extracted, word2vec and N-grams. N-grams include unigrams, bigrams, trigrams and four-grams. Using a random balanced split on the dataset, one Random Forest is trained for each type of features. Next, the predicted probabilities from each model on the test set are aggregated, so as to calculate the final late fusion model predictions. These probabilities are not equally weighted in the code. Weights are individually calculated for each class based on the OOB error weighting scheme. The output of the code consists of the confusion matrix for each model, including the late fusion model. These matrices can be provided as input, in order to extract evaluation metrics, such as accuracy, F-score etc. The code developed within MULTISENSOR for the category-based classification module is available at: <https://github.com/MKLab-ITI/category-based-classification>.

3.3 Application to MULTISENSOR Use Cases and evaluation

In this Section, we present the results from the application of the proposed final classification framework to a dataset of news articles retrieved from the MULTISENSOR Central News Repository (CNR).

3.3.1 Evaluation data

The dataset¹¹ used for the experiments within MULTISENSOR is an updated version of the corresponding MULTISENSOR dataset used for the experiments in D4.1 (see Section 4.4.2 – D4.1). The updated MULTISENSOR dataset contains 12,073 news articles, annotated to the

¹⁰ <https://www.python.org/>

¹¹ The dataset is available at: http://mklab.itl.gr/files/MULTISENSOR_NewsArticlesData_12073.7z

set of topics defined for the MULTISENSOR use cases, namely “Economy&finance&business”, “Health”, “Lifestyle&leisure”, “Nature&environment”, “Politics” and “Science&technology”. The numbers of news articles for each topic are listed below:

- **Economy&finance&business:** 3,805 news articles
- **Health:** 334 news articles
- **Lifestyle&leisure:** 3,742 news articles
- **Nature&environment:** 1,172 news articles
- **Politics:** 583 news articles
- **Science&technology:** 2,437 news articles

Feature extraction

For the process of extracting word2vec feature vectors and N-gram textual features from the news articles, we considered only the textual description of each news article.

Word2vec models were trained on a corpus of 3 million English news sentences downloaded from the Leipzig Corpora Collection¹². For each word contained in each news article, a feature vector was extracted from the trained word2vec model. In order to produce an overall word2vec representation for a given article, the corresponding feature vectors of the words contained in the article were simply averaged.

N-gram textual features were extracted globally from the MULTISENSOR dataset and not per topic. Four groups of N-grams were considered, namely unigrams, bigrams, trigrams and four-grams. When building the N-grams vocabulary and in order to avoid extracting an unnecessarily large number of N-grams, we ignored terms that had a document frequency lower than a given threshold. Specifically, for each group of N-grams the following threshold values (the values correspond to proportions of documents) were set: 0.05 for unigrams, 0.02 for bigrams and trigrams and 0.01 for four-grams.

Experimental setup

We applied a random balanced split to the MULTISENSOR dataset, meaning that the random splitting occurred within each class and preserved the overall class distribution of the data. Approximately 2/3 of the cases were used for training the RF models, while the rest (1/3) were used as test set, in order to estimate the performance of the classification models.

Regarding word2vec, in order to obtain the best parameter values for the dimensionality of the vectors and the width of the context window, we followed a grid search approach, coupled with 2-fold cross-validation on the training set. The optimal values, in terms of accuracy, were 200 for the number of dimensions and 12 for the context window width.

The RF parameter values were set in the following way: Regarding the number of trees used for constructing each RF model, we experimented with a gradually increasing number and computed the OOB error estimate for each experiment. We noticed that after using 1000 trees, the OOB error was stabilized. Therefore, the number of trees was set to $T=1000$. For each node split during the growing of a tree, the number of the subset of variables used to determine the best split was set to $p = \sqrt{k}$ (according to (Breiman, 2001)), where k is the total number of features of the dataset.

¹² <http://corpora2.informatik.uni-leipzig.de/download.html>

Finally, for the evaluation of the experiments, the precision, recall and F-score measures for each category were computed, along with their corresponding macro-averaged values and the accuracy on the entire test set (all categories included).

3.3.2 Evaluation results

In Table 4, the test set results from the application of RF to each modality can be seen. We notice that the N-gram modality yields better results than the word2vec modality (in terms of F-score values) in all topics, with the exception of the “Politics” topic, where word2vec achieves an F-score of 69.9%, compared to 62.1% for N-gram. In general, though, it can be said that both modalities perform comparably, with quite high accuracy values (83.3% for the N-gram RF model and 79.8% for word2vec) and macro-averaged F-scores (76.8% for N-gram and 74.2% for word2vec). That was not the case with the experiments conducted in D4.1, where the N-gram modality significantly outperformed the visual modality in all aspects (see Section 4.4.2 – D4.1). This justifies the selection of word2vec as the second modality for our experiments, since it can be considered more suitable for the classification task, compared to the visual modality.

Topics \ Modality	N-gram			Word2vec		
	Prec.	Rec.	F-score	Prec.	Rec.	F-score
Economy-Business-Finance	77.6%	95.5%	85.6%	80.5%	89.1%	84.6%
Health	90.4%	47.0%	61.8%	76.4%	42.0%	54.2%
Lifestyle-Leisure	87.8%	85.7%	86.7%	79.3%	84.8%	82.0%
Nature-Environment	90.1%	82.7%	86.2%	86.3%	80.4%	83.2%
Politics	94.2%	46.3%	62.1%	79.6%	62.3%	69.9%
Science-Technology	82.6%	74.6%	78.4%	76.3%	66.6%	71.1%
Macro-Average	87.1%	72.0%	76.8%	79.7%	70.9%	74.2%
Accuracy	83.3%			79.8%		

Table 4: Test set results from RF application to each modality

Tables 5 to 7 contain the test set results based on the late fusion of the RF models of the two modalities, using the OOB error estimate, the proximity ratio and the adjusted proximity ratio weighting strategy, respectively. In addition, the weight values assigned to each modality's RF model for each class, according to each weighting scheme, are depicted in the right side of each Table.

<div> <div>Weighting method</div> <div>Topics</div> </div>	N-gram + Word2vec (Weighting based on OOB error per topic)			Weight values assigned to each modality	
	Prec.	Rec.	F-score	N-gram	Word2vec
Economy- Business- Finance	80.1%	94.0%	86.5%	0.52	0.48
Health	86.8%	46.0%	60.1%	0.444	0.556
Lifestyle- Leisure	85.8%	87.8%	86.8%	0.505	0.495
Nature- Environment	88.6%	83.8%	86.1%	0.507	0.493
Politics	92.7%	58.3%	71.6%	0.431	0.569
Science- Technology	82.5%	72.1%	76.9%	0.539	0.461
Macro-average	86.1%	73.7%	78.0%		
Accuracy	83.6%				

Table 5: Test set results after the late fusion of RF regarding the OOB error weighting scheme

In general, for all examined weighting approaches, we notice a very slight improvement with respect to the accuracy measure, compared to the N-gram RF model (which is treated as the baseline approach in our experiments). Specifically, the OOB error weighting scheme yields an accuracy value of 83.6%, while the proximity ratio and adjusted proximity ratio approaches achieve accuracy values of 83.4% and 83.5%, respectively. According to the quality metrics established in deliverable D1.2 (Self-assessment plan v2) for category-based classification, we are more interested in the F-score measure (which considers both precision and recall). In this regard, the three late fusion schemes provide some improvement, in comparison with the baseline approach (76.8% macro-averaged F-score for

the baseline RF model, 1.2% improvement with the OOB error estimate and adjusted proximity ratio schemes and 1.4% improvement with the proximity ratio weighting approach). This improvement, although not as high as the lowest expectation for the D1.2 indicator of T4.1 (3%), can still be considered satisfactory, taking into account the fact that the baseline approach (N-gram RF model) performs considerably well in the first place.

The test set confusion matrices for all RF models can be found in Appendix A.

Topics \ Weighting method	N-gram + Word2vec (Weighting based on proximity ratio per topic)			Weight values assigned to each modality	
	Prec.	Rec.	F-score	N-gram	Word2vec
Economy-Business-Finance	80.6%	93.0%	86.3%	0.34	0.66
Health	88.9%	48.0%	62.3%	0.44	0.56
Lifestyle-Leisure	86.0%	87.7%	86.9%	0.71	0.29
Nature-Environment	88.3%	83.2%	85.7%	0.62	0.38
Politics	88.2%	60.0%	71.4%	0.27	0.73
Science-Technology	80.8%	72.4%	76.3%	0.4	0.6
Macro-average	85.5%	74.1%	78.2%		
Accuracy	83.4%				

Table 6: Test set results after the late fusion of RF regarding the proximity ratio weighting scheme

<div> <div>Weighting method</div> <div>Topics</div> </div>	N-gram + Word2vec (Weighting based on adjusted proximity ratio per topic)			Weight values assigned to each modality	
	Prec.	Rec.	F-score	N-gram	Word2vec
Economy-Business-Finance	80.5%	93.2%	86.4%	0.36	0.64
Health	87.0%	47.0%	61.0%	0.38	0.62
Lifestyle-Leisure	86.3%	87.6%	87.0%	0.71	0.29
Nature-Environment	88.3%	83.2%	85.7%	0.63	0.37
Politics	87.5%	60.0%	71.2%	0.22	0.78
Science-Technology	81.1%	72.6%	76.6%	0.44	0.56
Macro-average	85.1%	73.9%	78.0%		
Accuracy	83.5%				

Table 7: Test set results after the late fusion of RF regarding the adjusted proximity ratio weighting scheme

4 TOPIC-EVENT DETECTION

Topic-event detection in news articles is a very important problem for journalists and media monitoring companies, because of their need to quickly detect interesting articles. This problem becomes also very challenging and complex, given the relatively large amount of news articles produced on a daily basis. In general, the topic detection task aims to group together stories-documents that discuss about the same topic-event. Formally, a topic is defined by Allan (2002) as *“a specific thing that happens at a specific time and place along with all necessary preconditions and unavoidable consequences”*. Allan (2002) clarified that the notion of “topic” is not general like “accidents” but is limited to a specific collection of related events of the type accident, such as “cable car crash”. We shall refer to topics-events simply as topics or as news clusters.

The challenges of the aforementioned problem are summarized into two main directions: (a) discover the correct number of topics and (b) group the most similar news articles into clusters. We face these challenges under the following assumptions. Firstly, we take into account that real data is highly noisy and the number of clusters is not known. Secondly, we assume that there is a lower bound for the minimum number of documents per news cluster. Thirdly, we consider the names/labels of the topics unknown.

Towards addressing this problem, we introduce a novel hybrid clustering framework for topic detection, which combines automatic estimation of the number of clusters and assignment of news articles into topics of interest. The estimation of the number of clusters is done by our novel “DBSCAN-Martingale” method, which can deal with the aforementioned assumptions. The main idea is to progressively extract all clusters (extracted by a density-based algorithm) by applying Doob’s martingale and then apply a well-established method for the assignment of news articles to topics, such as Latent Dirichlet Allocation. The proposed hybrid framework does not consider known the number of topics to be discovered, but requires only the more intuitive parameter *minPts*, as a lower bound for the number of documents per topic. Each realization of the DBSCAN-Martingale provides the number of detected topics and due to randomness this number is a random variable. As the final number of detected topics, we use the majority vote over 10 realizations of the DBSCAN-Martingale.

Our contribution is summarized as follows:

- We present our novel DBSCAN-Martingale process, which progressively estimates the number of clusters in a dataset.
- We introduce a novel hybrid news clustering framework, which combines our DBSCAN-Martingale with Latent Dirichlet Allocation.

In the following, in Section 4.1 we present existing approaches for the topic detection problem, for news clustering and for density-based clustering. In Section 4.2 the MULTISENSOR framework for topic detection is discussed, which uses the method “DBSCAN-Martingale” for the estimation of the number of topics (Section 4.3). The topic detection module is reported in Section 4.4 and finally, in Section 4.5 we test both our novel method for the estimation of the number of clusters and the hybrid topic detection framework in several datasets of various sizes.

4.1 Relevant work on topic-event detection and news clustering

Traditionally, the topic detection task is tackled as a clustering problem (Aggarwal and Zhai, 2012), due to the absence of training sets. The clustering task usually involves feature selection (Qian and Zhai, 2014), spectral clustering (Kumar and Daume, 2011) and k-means oriented (Aggarwal and Zhai, 2012) techniques, assuming mainly that the number of topics to be discovered is known a priori and there is no noise, i.e. news items which do not belong to any of the news clusters. However, MULTISENSOR considers the more general and realistic case, where the number of topics to be discovered is unknown and it is possible to have irrelevant news articles.

Latent Dirichlet Allocation (LDA) is a popular model for topic modelling, given the number of topics k (Blei et al, 2003). LDA has been generalized to nonparametric Bayesian approaches, such as the hierarchical Dirichlet process (Teh et al., 2006) and DP-means (Kulis and Jordan, 2012), which predict the number of topics k . The extraction of the correct number of topics is equivalent to the estimation of the correct number of clusters in a dataset. The majority vote among 30 clustering indices has been proposed in (Charrad et al., 2014) as an indicator for the number of clusters in a dataset. In contrast, we propose an alternative majority vote among 10 realizations of the “DBSCAN-Martingale”, which is a modification of the DBSCAN algorithm (Ester et al., 1996) and has three main advantages and characteristics: (a) they discover clusters with not-necessarily regular shapes, (b) they do not require the number of clusters and (c) they extract noise. The parameters of DBSCAN are the density level ε and a lower bound for the minimum number of points per cluster.

Other approaches for clustering that could be applied to topic detection, without knowing the number of topics, involves density-based clustering algorithms. The graph-analogue of DBSCAN has been presented in (Campellon et al., 2013) and dynamically adjusting the density level ε , the nested hierarchical sequence of clusterings results to the HDBSCAN algorithm (Campello et al., 2013). OPTICS (Ankerst et al., 1999) is very useful for the visualization of the cluster structure and for the optimal selection of the density level ε . The OPTICS reachability plot allows for determining the number of clusters in a dataset by counting the “dents” of the OPTICS diagram. F-OPTICS (Schneider and Vlachos, 2013) has reduced the computational cost of the OPTICS algorithm using a probabilistic approach of the reachability distance, without significant accuracy reduction. The OPTICS- ξ algorithm (Ankerst et al., 1999) requires an extra parameter ξ , which has to be manually set in order to find “dents” in the OPTICS reachability plot. The automatic extraction of clusters from the OPTICS reachability plot, as an extension of the OPTICS- ξ algorithm, has been presented in (Sander et al., 2003) and has been outperformed by HDBSCAN (Campello et al., 2013) in several datasets of any nature. The recent work of Mai et al. (2014) utilizes lower bounding functions in their “A-DBSCAN” algorithm to approximate quickly the results of DBSCAN. In the context of news clustering, we shall examine whether some of these density-based algorithms perform well on the topic detection problem and we shall compare them with our DBSCAN-Martingale in terms of the number of estimated clusters. Contrary to DBSCAN, the DBSCAN-Martingale regards the density level ε as a random variable and the clusters are progressively extracted. All the abovementioned methods, which do not require a priori known the number of clusters, are combined with LDA in order to examine whether the use of DBSCAN-Martingale (combined with LDA) provides the most efficient assignment of news articles to clusters.

4.2 The MULTISENSOR topic detection framework based on multimodal features

The MULTISENSOR framework for topic detection is approached as a news clustering problem, where the number of topics needs to be estimated. The overall framework is language independent because it is based on language-agnostic textual features and is presented in Figure 11. The number of topics k is estimated by our proposed DBSCAN-Martingale, which is introduced in Section 4.3, and the assignment of news articles to topics is done using Latent Dirichlet Allocation (LDA).

The MULTISENSOR topic detection framework is a hybrid framework based on LDA and DBSCAN-Martingale, which is a density-based clustering method. LDA performs well on text clustering but requires as input the number of clusters. On the other hand, density-based clustering algorithms do not require the number of clusters, but their performance in text clustering is limited, when compared to LDA.

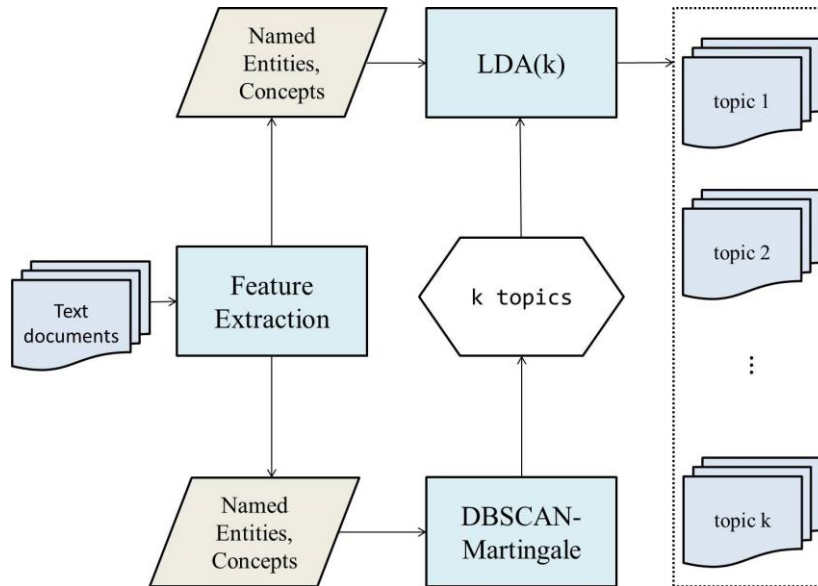


Figure 11: The MULTISENSOR topic detection framework using DBSCAN-Martingale and LDA

LDA is a probabilistic topic model, which assumes a Bag-of-Words (Bow) representation of the collection of documents. Each topic is a distribution over terms in a fixed vocabulary, which assigns probabilities to words. Moreover, LDA assumes that documents exhibit multiple topics and assigns a probability distribution on the set of documents. Finally, LDA assumes that the order of words does not matter and, therefore, LDA is not applicable to word n -grams for $n \geq 2$. We refer to word n -grams as “uni-grams” for $n = 1$ and as “bi-grams” for $n = 2$. Following the concept of “phrase extraction” (Aggarwal and Zhai, 2012), DBSCAN-Martingale performs well on the bi-grams, after stemming by Porter’s algorithm (Porter, 1980). However, in the case of a multilingual corpus, uni-grams and bi-grams cannot be used, if not translated. MULTISENSOR applies the DBSCAN-Martingale and LDA on the language independent concepts and named entities, as shown in Figure 11.

4.3 Estimation of the number of topics using the DBSCAN-Martingale

The present Section discusses the construction of DBSCAN-Martingale. Firstly, we provide the necessary background in density-based clustering and the notation which we adopt. Secondly, we progressively estimate the number of clusters in a dataset, by defining a stochastic process, which is then shown to be a Martingale process.

4.3.1 Notation and background on DBSCAN and OPTICS

Given a dataset of n -points, news articles in our case, density-based clustering algorithms provide as output the clustering vector C with values the cluster IDs $C[j]$ for each instance $j = 1, 2, \dots, n$. The j -th element of a vector C is denoted by $C[j]$. For example, if the j -th point belongs to cluster k , the clustering vector C has $C[j] = k$. In case the j -th document is marked as noise, then $C[j] = 0$. We denote by $C_{DBSCAN(\varepsilon)}$ the clustering vector provided by the DBSCAN algorithm for the density level ε . Low values of ε result to $C_{DBSCAN(\varepsilon)}[j] = 0$, for all j (all points are marked as noise). On the other hand, high values of ε result to $C_{DBSCAN(\varepsilon)}[j] = 1$, for all j (all points are assigned to cluster 1). If a clustering vector has only zeros and ones, one cluster has been detected and the partitioning is trivial.

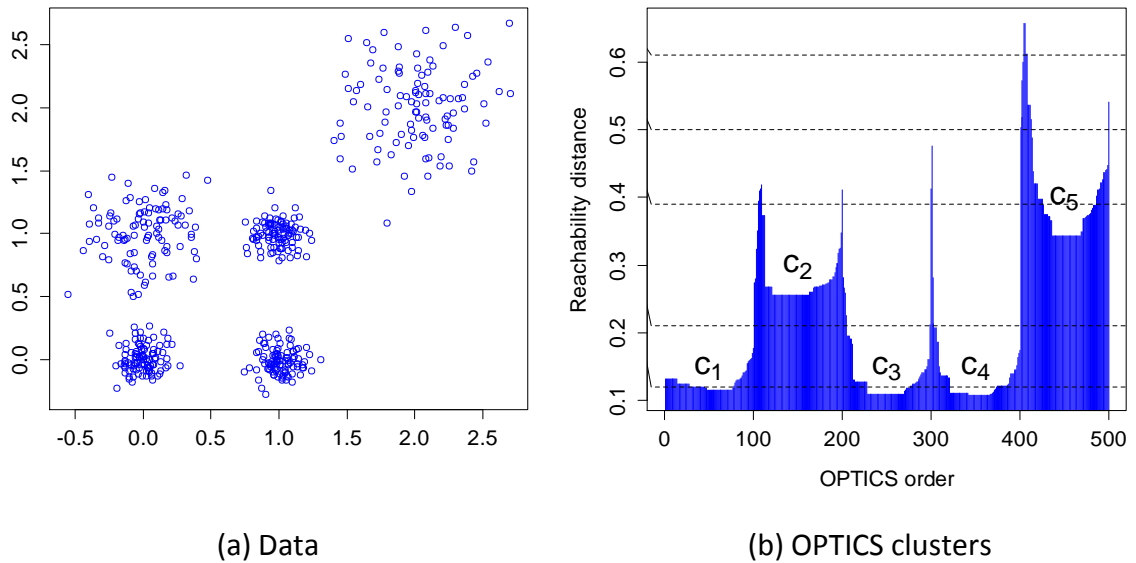


Figure 12: OPTICS reachability plot and randomly generated density levels

Clusters detected by DBSCAN strongly depend on the density level ε . An indicative example is shown in Figure 12, where the 5 clusters C_1, C_2, C_3, C_4, C_5 do not have the same density and it is evident that there is no single value of ε that can output all clusters. In Figure 12(b), we illustrate the OPTICS reachability plot of the data in Figure 12(a), with 5 randomly selected density levels (horizontal dashed lines) and none of them is able to extract all corresponding to the synthetic dataset of Figure 12(a).

In order to deal with this problem, we introduce an extension of DBSCAN based on Doob's Martingale, which allows for introducing a random variable ε and involves the construction of a Martingale process, which progressively approaches the clustering vector with all clusters included.

4.3.2 Estimation of the number of clusters with the DBSCAN-Martingale

We introduce a probabilistic method to estimate the number of clusters, by constructing a Martingale stochastic process (Doob, 1953). The martingale construction is based on Doob's martingale (Doob, 1953), in which we progressively gain knowledge about the result of a random variable. In the density-based clustering problem, the random variable that needs to be known is the vector of cluster IDs, which is a combination of T clustering vectors $C_{DBSCAN(\varepsilon_t)}, t = 1, 2, \dots, T$.

First, we generate a sample of size T with random numbers $\varepsilon_t, t = 1, 2, \dots, T$ uniformly in $[0, \varepsilon_{max}]$, where ε_{max} is an upper bound for the density levels. The sample of $\varepsilon_t, t = 1, 2, \dots, T$ is sorted in increasing order and the values of ε_t can be demonstrated on an OPTICS reachability plot, as shown in Figure 12(b) for $T = 5$. For each density level ε_t we find the corresponding clustering vectors $C_{DBSCAN(\varepsilon_t)}$ for all stages $t = 1, 2, \dots, T$.

In the beginning of the algorithm, there are no clusters detected. In the first stage ($t = 1$), all clusters detected by $C_{DBSCAN(\varepsilon_1)}$ are kept, corresponding to the lowest density level ε_1 . In the second stage ($t = 2$), some of the detected clusters by $C_{DBSCAN(\varepsilon_2)}$ are new and some of them have also been detected at previous stage ($t = 1$). In order to keep only the newly detected clusters of the second stage ($t = 2$), we keep only groups of numbers of the same cluster ID with size greater than $minPts$. An example with two iterations of the process is demonstrated in Figure 13, where we adopted the notation X^T for the transpose of the matrix or vector X .

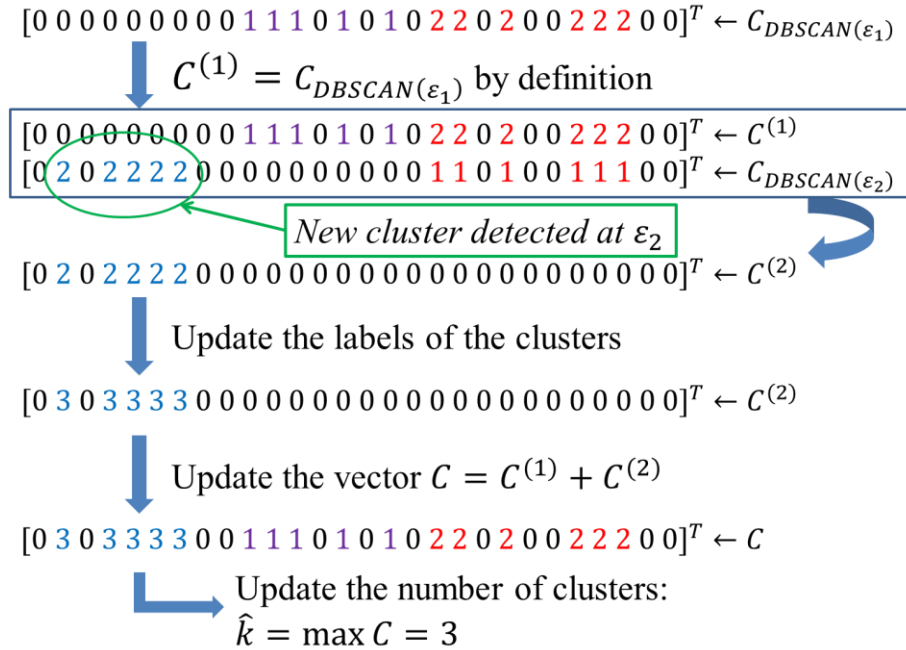


Figure 13: One realization of the DBSCAN-Martingale with $T = 2$ iterations. The points with cluster label **2** in $C^{(1)}$ are re-discovered as a new cluster by $C_{DBSCAN(2)}$ but the update rule keeps only the newly detected cluster

Formally, we define the sequence of vectors $C^{(t)}, t = 1, 2, \dots, T$, where $C^{(1)} := C_{DBSCAN(\varepsilon_1)}$,

$$C^{(t)}[j] := \begin{cases} 0 & \text{if point } j \text{ belongs to a previously extracted cluster} \\ C_{DBSCAN(\varepsilon_t)}[j] & \text{otherwise} \end{cases} \quad (4.1)$$

Since the stochastic process $C^{(t)}, t = 1, 2, \dots, T$ is a Martingale, as shown in Section 4.3.3, and, since each $C_{DBSCAN(\varepsilon_t)}, t = 1, 2, \dots, T$ is the output of DBSCAN for the density level ε_t , we call our method “DBSCAN-Martingale”.

Finally, we relabel the cluster IDs. Assuming that r clusters have been detected for the first time at stage t , we update the cluster labels of $C^{(t)}$ starting from $1 + \max_j C^{(t-1)}[j]$, to $r + \max_j C^{(t-1)}[j]$. Note that the maximum value of a clustering vector coincides with the number of clusters.

The sum of all vectors $C^{(t)}$ up to stage T is the final clustering vector of our algorithm:

$$C = C^{(1)} + C^{(2)} + \dots + C^{(T)} \quad (4.2)$$

The estimated number of clusters \hat{k} is the maximum value of the final clustering vector C :

$$\hat{k} = \max_j C[j] \quad (4.3)$$

The process we have formulated, namely the DBSCAN-Martingale, is represented as pseudo code in Algorithm 1. Algorithm 1 extracts clusters sequentially, combines them into one single clustering vector and outputs the most updated estimation of the number of clusters \hat{k} .

Algorithm 1 DBSCAN-Martingale(*minPts*) **return** \hat{k}

```

1:  Generate a random sample of  $T$  values in  $[0, \varepsilon_{max}]$ 
2:  Sort the generated sample  $\varepsilon_t, t = 1, 2, \dots, T$ 
3:  for  $t = 1$  to  $T$ 
4:    find  $C_{DBSCAN(\varepsilon_t)}$ 
5:    compute  $C^{(t)}$  as in Eq. (4.1)
6:    update the cluster IDs
7:    update the final clustering vector as in Eq. (4.2)
8:    update  $\hat{k} = \max_j C[j]$ 
9:  end for
10: return  $\hat{k}$ 

```

The DBSCAN-Martingale requires T iterations of the DBSCAN algorithm, which runs in $\mathcal{O}(n \log n)$ if a tree-based spatial index can be used and in $\mathcal{O}(n^2)$ without tree-based spatial indexing (Ankerst et al., 1999). Therefore, the DBSCAN-Martingale runs in $\mathcal{O}(Tn \log n)$ for tree-based indexed datasets and in $\mathcal{O}(Tn^2)$ without tree-based indexing. Our code (Section 4.4) is written in R¹³, using the `dbscan`¹⁴ package, which runs DBSCAN in $\mathcal{O}(n \log n)$ with kd-tree data structures for fast nearest neighbor search.

The DBSCAN-Martingale (one execution of Algorithm 1) is illustrated for example, on the OPTICS reachability plot of Figure 12(b) where, for the random sample of density levels $\varepsilon_t, t = 1, 2, \dots, 5$, (horizontal dashed lines), we sequentially extract all clusters. In the first density level $\varepsilon_1 = 0.12$, DBSCAN-Martingale extracts the clusters C_1, C_3 and C_4 , but in the

¹³ <https://www.r-project.org/>

¹⁴ <https://cran.r-project.org/web/packages/dbscan/index.html>

density level $\varepsilon_2 = 0.21$ no new clusters are extracted. In the third density level, $\varepsilon_3 = 0.39$, the clusters C_2 and C_5 are added to the final clustering vector and in the other density levels, ε_4 and ε_5 there are no new clusters to extract. The number of clusters extracted up to stage t is shown in Figure 14. Observe that at $t = 3$ iterations, DBSCAN-Martingale has output $k = 5$ and for all iterations $t > 3$ there are no more clusters to extract.

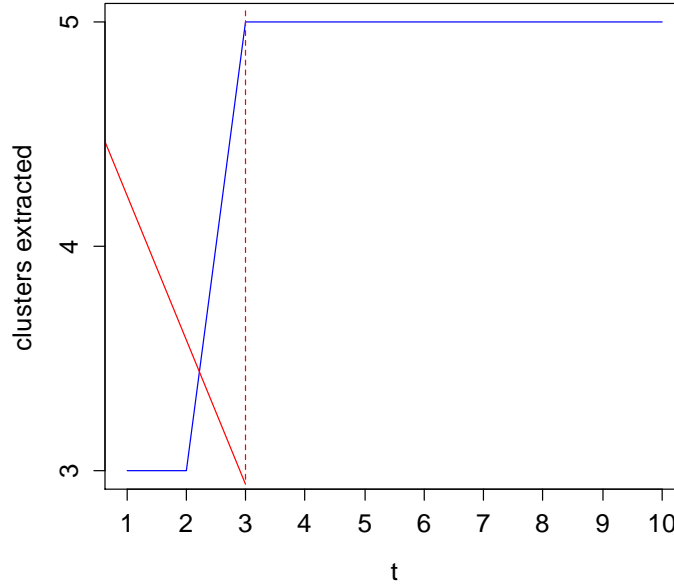


Figure 14: The convergence of DBSCAN-Martingale on the OPTICS plot of Figure 12

The estimation of number of clusters \hat{k} is a random variable, because it inherits the randomness of the density levels $\varepsilon_t, t = 1, 2, \dots, T$. For each execution of Algorithm 1, one realization of the DBSCAN-Martingale generates \hat{k} , so we propose as the final estimation of the number of clusters the majority vote over 10 realizations of the DBSCAN-Martingale.

Algorithm 2 MajorityVote(*realizations*, *minPts*) **return** \hat{k}

```

1:  Set clusters =  $\emptyset, k = 0$ 
2:  for  $r = 1$  to realizations
3:     $k = \text{DBSCAN-Martingale}(\text{minPts})$ 
4:    clusters = AppendTo(clusters,  $k$ )
5:  end for
6:   $\hat{k} = \text{mode}(\text{clusters})$ 
7:  return  $\hat{k}$ 

```

Algorithm 2 outputs the majority vote over a fixed number of realizations of the DBSCAN-Martingale. For each realization, the estimated number of clusters k is added to the list *clusters* and the majority vote is obtained from the mode of *clusters*, since the mode is defined as the most frequent value in a list. The percentage of realizations where the DBSCAN-Martingale outputs exactly \hat{k} clusters is a probability distribution, such as the one shown in Figure 15, which corresponds to the illustrative dataset of Figure 12. We note that

the same result $\hat{k} = 5$ appears for a wide range of the parameter $minPts$, in Figure 15(b), a fact that demonstrates the robustness of our approach.

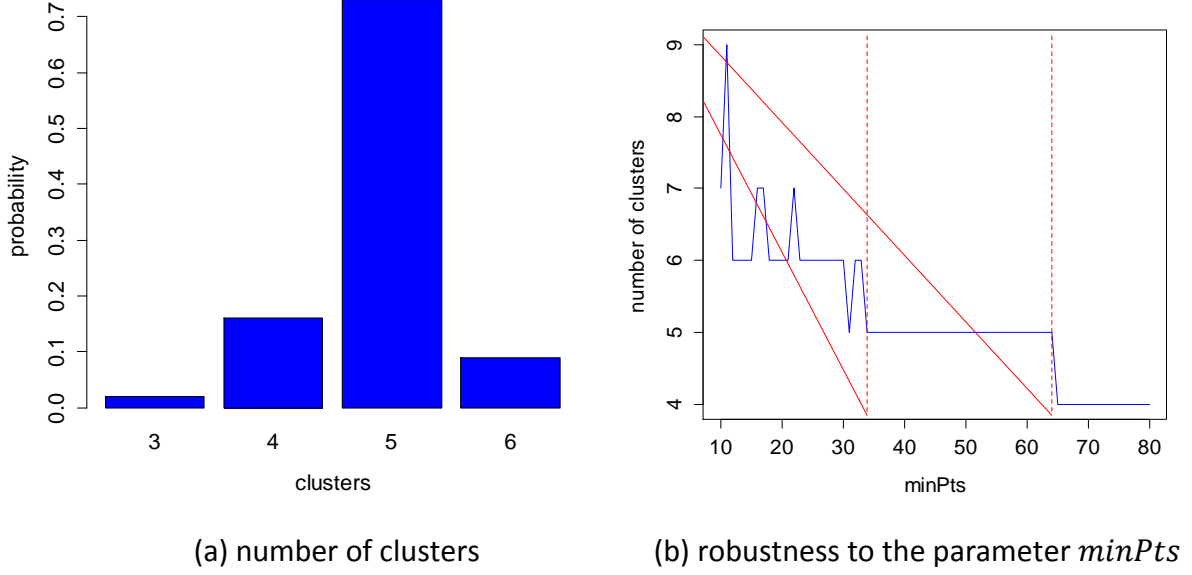


Figure 15: The number of clusters as generated by DBSCAN-Martingale after 100 realizations for $minPts = 50$

4.3.3 The martingale stochastic process

In this Section it is proved that the sequence of vectors $C^{(t)}$, Equation (4.1), is a martingale process. Martingale is a random process X_1, X_2, \dots for which the expected future value of X_{t+1} , given all prior values X_1, X_2, \dots, X_t , is equal to the present observed value X_t . Doob's martingale is a generic martingale construction, in which our knowledge about a random variable is progressively obtained. The conditional expectation of a random variable X given Y is denoted by $E[X|Y]$.

Definition 1. [Doob's Martingale] (Doob, 1953). Let X, Y_1, Y_2, \dots be any random variables with $E[|X|] < \infty$. Then, if X_t is defined by $X_t = E[X|Y_1, Y_2, \dots, Y_t]$, the sequence of $X_t, t = 1, 2, \dots$ is a martingale.

In this context, we will show that the sequence of clustering vectors

$$X_t = C^{(1)} + C^{(2)} + \dots + C^{(t)}, \quad t = 1, 2, \dots, T$$

is Doob's martingale for the sequence of random variables $Y_t = C_{DBSCAN(\epsilon_t)}, t = 1, 2, \dots, T$.

We denote by $\langle Z_i, Z_l \rangle = \sum_j Z_i[j] \cdot Z_l[j]$ the inner product of any two vectors Z_i and Z_l , so as to prove the following Lemma.

Lemma 1. *If two clustering vectors Z_i, Z_l are mutually orthogonal, they contain different clusters.*

Proof. The values of the clustering vectors are cluster IDs so they are non-negative integers. Points which do not belong to any of the clusters (noise) are assigned zeros. Since $\langle Z_i, Z_l \rangle$

$= \sum_{j=1}^n Z_i[j] \cdot Z_l[j] = 0$ and based on the fact that when a sum of non-negative integers is zero, then all integers are zero, we obtain $Z_i[j] = 0$ OR $Z_l[j] = 0$ for all $j = 1, 2, \dots, n$.

For example, the clustering vectors

$$Z_i = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

$$Z_l = [1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 2 \ 2 \ 2 \ 2 \ 2]^T$$

are mutually orthogonal and contain different clusters.

Martingale Construction

Each density level $\varepsilon_t, t = 1, 2, \dots, T$ provides one clustering vector $C_{DBSCAN(\varepsilon_t)}$ for all $t = 1, 2, \dots, T$. As t increases, more clustering vectors are computed and we gain knowledge about the final clustering vector C .

In Equation (4.1), we constructed a sequence of vectors $C^{(t)}, t = 1, 2, \dots, T$, where each $C^{(t)}$ is orthogonal to all $C^{(1)}, C^{(2)}, \dots, C^{(t-1)}$ from Lemma 1. The sum of all clustering vectors $C^{(1)} + C^{(2)} + \dots + C^{(t-1)}$ has zeros as cluster IDs in the points which belong to the clusters of $C^{(t)}$. Therefore, $C^{(t)}$ is also orthogonal to $C^{(1)} + C^{(2)} + \dots + C^{(t-1)}$. Using the orthogonality, the vector $C^{(1)} + C^{(2)} + \dots + C^{(t)}$ is our “best prediction” for the final clustering vector C at stage t . The expected final clustering vector at stage t is:

$$E[C | C_{DBSCAN(\varepsilon_1)}, C_{DBSCAN(\varepsilon_2)}, \dots, C_{DBSCAN(\varepsilon_t)}] = C^{(1)} + C^{(2)} + \dots + C^{(t)}$$

Initially, the final clustering C vector is the zero vector O . Our knowledge about the final clustering vector up to stage t is restricted to $C^{(1)} + C^{(2)} + \dots + C^{(t)}$ and finally, at stage $t = T$, we have gained all available knowledge about the final clustering vector C :

$$C = E[C | C_{DBSCAN(\varepsilon_1)}, C_{DBSCAN(\varepsilon_2)}, \dots, C_{DBSCAN(\varepsilon_T)}]$$

As a result, after T stages, all clusters are extracted and the number of clusters is the maximum of the cluster IDs of C .

4.4 Topic detection module

The topic detection module is based on the DBSCAN-Martingale, which has been developed in R¹⁵, version 3.2.3, and requires the “dbscan” R package (Hahsler, 2015). The input is a data matrix to be clustered. The output is a probability distribution over the number of clusters and a barplot, showing the number of clusters which is more probable to describe the optimal partitioning of the dataset.

The topic detection module requires as input a folder, which contains a list of text documents. The output is a JSON file with a list of topics with labels provided by the concepts or named entities that have the highest probability within each topic. The requirements are the R packages “tm” (Feinerer and Hornik, 2015), “dbscan” (Hahsler, 2015), “topicmodels” (Gruen and Hornik, 2011) and “rjson” (Couture-Beil, 2014).

The output of the topic detection module can be visualized in Figure 16 for the query “home appliances”, where there are 267 retrieved results, which are then clustered by 9 topics. The

¹⁵ <https://www.r-project.org/>

font size of the clusters' labels depends on the particular word probability within each cluster.

The language button is left at the position "All", i.e. no language is specified for this task. The topic detection module is applied to the language independent textual features of each webpage.

The code developed within MULTISENSOR for the topic detection module is available at: <https://github.com/MKLab-ITI/topic-detection>. In the following, the MULTISENSOR topic detection module is tested in several news-oriented datasets.

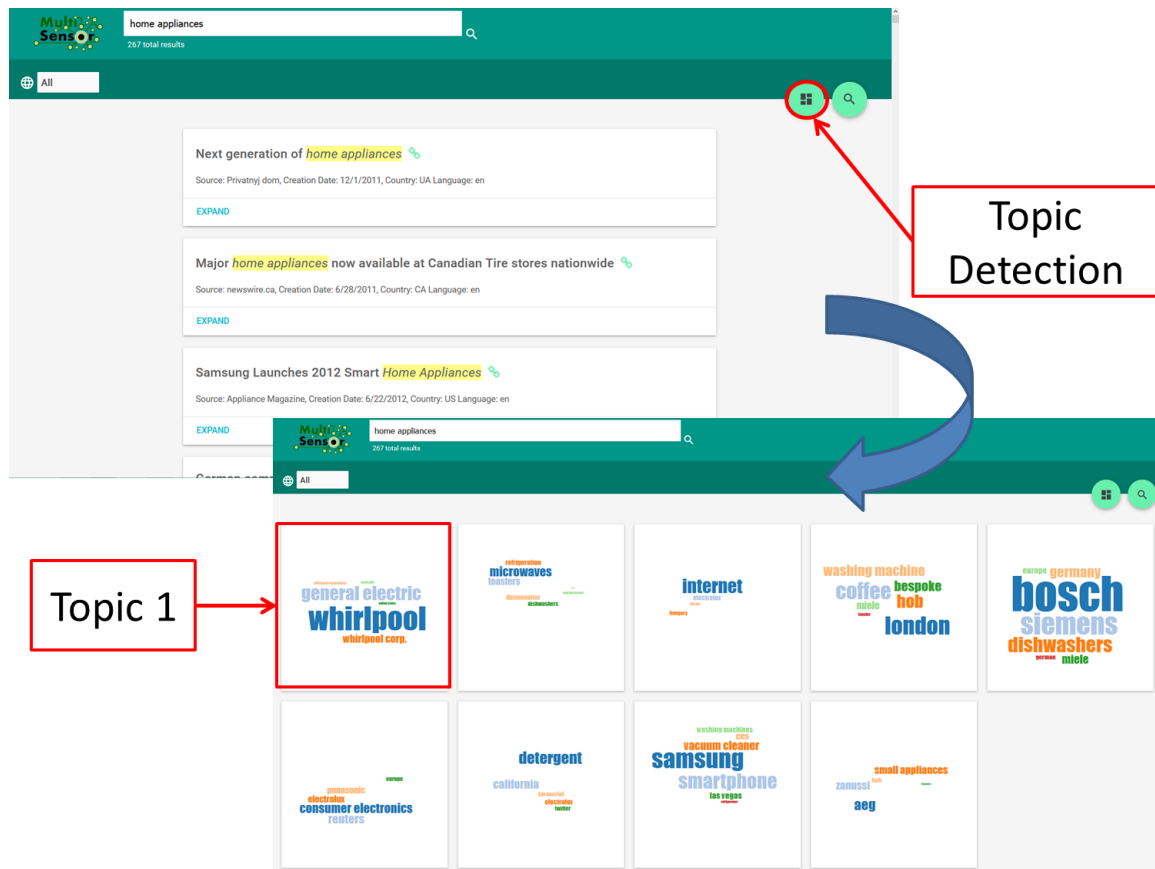


Figure 16: Demonstration of the topic detection module

4.5 Application to MULTISENSOR Use Cases and evaluation

The evaluation of our method is done in two levels. Firstly, we test whether the output of the majority vote over 10 realizations of the DBSCAN-Martingale matches the ground-truth number of clusters. Secondly, we evaluate the overall hybrid topic detection framework, using the estimated number of clusters and Latent Dirichlet Allocation. We also evaluate the MULTISENSOR topic detection module using language independent concepts and named entities.

4.5.1 Evaluation data

A part of the present MULTISENSOR database with 10,476 webpages in total was used for evaluation, in the context of multilingual topic detection. We use the retrieved results for a given query in order to cluster them into labelled clusters (topics) without knowing the number of clusters, as shown in Figure 16. The queries which are used are the following:

- energy crisis
- energy policy
- home appliances
- solar energy

Apart from the MULTISENSOR database, the DBSCAN-Martingale and the topic detection framework have been tested in the 20newsgroups-mini dataset with 2,000 articles, which is available on the UCI repository¹⁶ and in 220 news articles, which are references to specific Wikipedia pages so as to ensure reliable ground-truth: the WikiRef220. We also use two subsets of WikiRef220, namely the WikiRef186 and the WikiRef150, in order to test DBSCAN-Martingale in four datasets of sizes 2000, 220, 150 and 115 documents respectively.

The selected topics of the WikiRef220 dataset (and the number of articles per topic) are: Paris Attacks November 2015 (36), Barack Obama (5), Premier League (37), Cypriot Financial Crisis 2012-2013 (5), Rolling Stones (1), Debt Crisis in Greece (5), Samsung Galaxy S5 (35), Greek Elections June 2012 (5), smartphone (5), Malaysia Airlines Flight 370 (39), Stephen Hawking (1), Michelle Obama (38), Tohoku earthquake and tsunami (5), NBA draft (1), U2 (1), Wall Street (1). The topics Barack Obama, Cypriot Financial Crisis 2012-2013, Rolling Stones, Debt Crisis in Greece, Greek Elections June 2012, smartphone, Stephen Hawking, Tohoku earthquake and tsunami, NBA draft, U2 and Wall Street appear no more than 5 times and therefore, they are regarded as noise. The remaining 5 topics of WikiRef220 are:

- Paris Attacks November 2015¹⁷
- Premier League¹⁸
- Malaysia Airlines Flight 370¹⁹
- Samsung Galaxy S5²⁰
- Michelle Obama²¹.

The WikiRef186 dataset (4 topics) is the WikiRef220 without 34 documents related to “Malaysia Airlines Flight 370” and the WikiRef150 dataset (3 topics) is the WikiRef186 without the 36 documents related to “Paris Attacks”. We selected these datasets because we focus on datasets with news clusters which are event-oriented, like “Paris Attacks November 2016” or they discuss about specific topics like “Barack Obama” (rather than “Politics” in general). The news-articles categorization problem is a supervised classification

¹⁶ <http://archive.ics.uci.edu/ml/datasets.html>

¹⁷ https://en.wikipedia.org/wiki/November_2015_Paris_attacks

¹⁸ https://en.wikipedia.org/wiki/Premier_League

¹⁹ https://en.wikipedia.org/wiki/Malaysia_Airlines_Flight_370

²⁰ https://en.wikipedia.org/wiki/Samsung_Galaxy_S5

²¹ https://en.wikipedia.org/wiki/Michelle_Obama

problem, because training sets are available, contrary to the news clustering problem where, for example, the topic "Malaysia Airlines Flight 370" had no training set before the 8th of March 2014. We assume that 2,000 news articles is a reasonable upper bound for the number of recent or retrieved news articles that can be considered for clustering, in line with other datasets that were used to evaluate similar methods (Qian and Zhai, 2014; Campello et al., 2013).

In order to evaluate the 20newsgroups, the WikiRef220, the WikiRef186 and the WikiRef150 datasets in a more general approach, we use uni-grams and bi-grams, assuming a Bag-of-Words representation of text. Before the extraction of uni-grams and bi-grams, we remove the SMART²² stopwords list and we then stem the words using Porter's algorithm (Porter, 1980). Uni-grams are filtered out if they occur less than 6 times and bi-grams if they occur less than 20 times. The final bi-grams are normalized using tf-idf weighting and, in all datasets, the upper bound for the density level is taken $\varepsilon_{max} = 3$.

4.5.2 Evaluation results in public datasets

Using the abovementioned datasets, we firstly present the performance of DBSCAN-Martingale as a method for estimating the number of clusters. Afterwards, we evaluate the topic detection framework using the estimated number of clusters along with LDA.

Evaluation of the number of clusters

The estimation of the number of clusters is compared to other indices or methods, listed in Table 8, which either estimate the number of clusters directly, or provide a clustering vector without any knowledge of the number of clusters.

Index	Reference	WikiRef150	WikiRef186	WikiRef220	20news
Ground truth number of clusters		3	4	5	20
CH	Caliński and Harabasz, 1974	30	29	30	30
Duda	Duda and Hart, 1973	2	2	2	2
Pseudo t^2	Duda and Hart, 1973	2	2	2	2
C-index	Hubert and Levin, 1976	27	2	2	2
Ptbiserial	Milligan and Cooper, 1985	11	7	6	30
DB	Davies and Bouldin, 1979	2	4	6	2
Frey	Frey and Groenewoud 1972	2	2	2	5
Hartigan	Hartigan, 1975	18	20	16	24
Ratkowsky	Ratkowsky and Lance, 1978	20	24	29	30
Ball	Ball and Hall, 1965	3	3	3	3
McClain	McClain and Rao, 1975	2	2	2	2
KL	Krzanowski and Lai, 1988	14	15	17	15
Silhouette	Kaufman and Rousseeuw, 1990	30	4	4	2
Dunn	Dunn, 1974	2	4	5	3

²² <http://jmlr.csail.mit.edu/papers/volume5/lewis04a/a11-smart-stop-list/english.stop>

SDindex	Hal2000	4	4	6	3
SDbw	Halkidi et al., 2001	30	7	6	3
NbClust	Charrad et al., 2014	2	2	6	2
DP-means	Kulis and Jordan, 2012	4	4	7	15
HDBSCAN-EOM	Campello et al., 2013	5	5	5	36
DBSCAN-Martingale	MULTISENSOR	3	4	5	20

Table 8: Evaluation of the DBSCAN-Martingale in estimating the number of topics

The index “NbClust”, which is computed using the NbClust²³ package, is the majority vote among the 24 indices: CH, Duda, Pseudot2, C-index, Beale, CCC, Ptbiserial, DB, Frey, Hartigan, Ratkowsky, Scott, Marriot, Ball, Trcovw, Tracew, Friedman, McClain, Rubin, KL, Silhouette, Dunn, SDindex, SDbw (Charrad et al., 2014). The Dindex and Hubert's I' are graphical methods and they are not involved in the majority vote. The indices GAP, Gamma, Gplus and Tau are not included in the majority vote, due to their high computational cost. The NbClust package requires as a parameter the maximum number of clusters to look for, which is set to 30. For the extraction of clusters from the HDBSCAN hierarchy, we adopt the EOM-optimization (Campello et al., 2013) and for the nonparametric method DP-means.

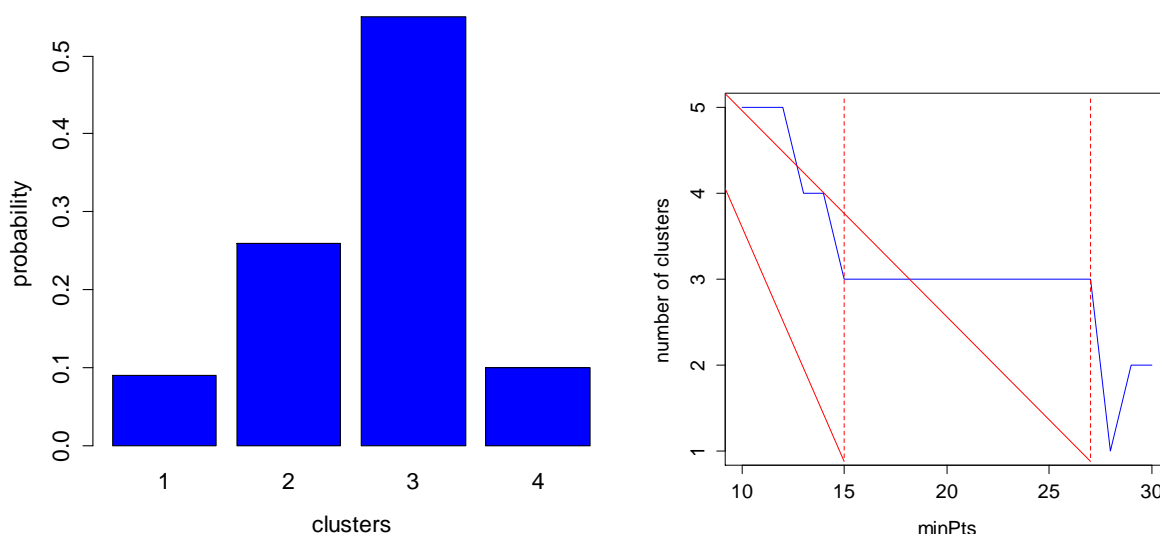


Figure 17: Number of topics estimation for the Wiki150 dataset ($minPts = 20$)

²³ <https://cran.r-project.org/web/packages/NbClust/index.html>

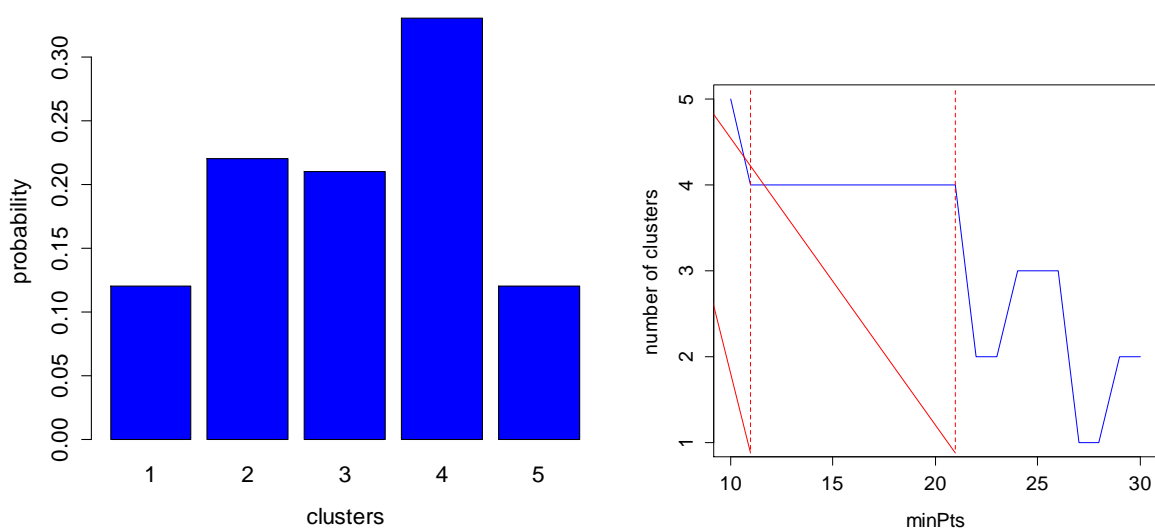


Figure 18: Number of topics estimation for the Wiki186 dataset ($minPts = 20$)

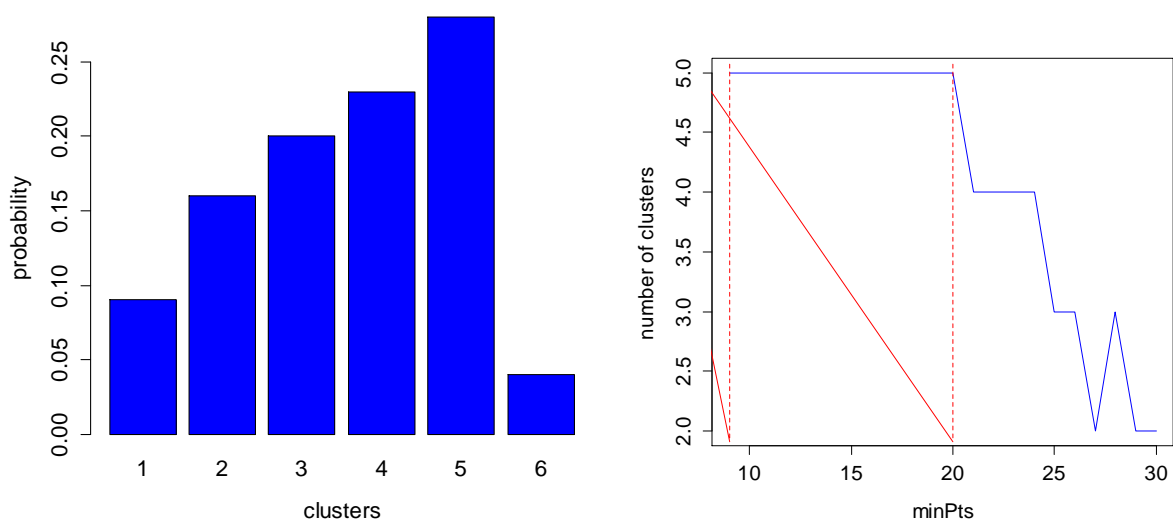


Figure 19: Number of topics estimation for the Wiki150 dataset ($minPts = 20$)

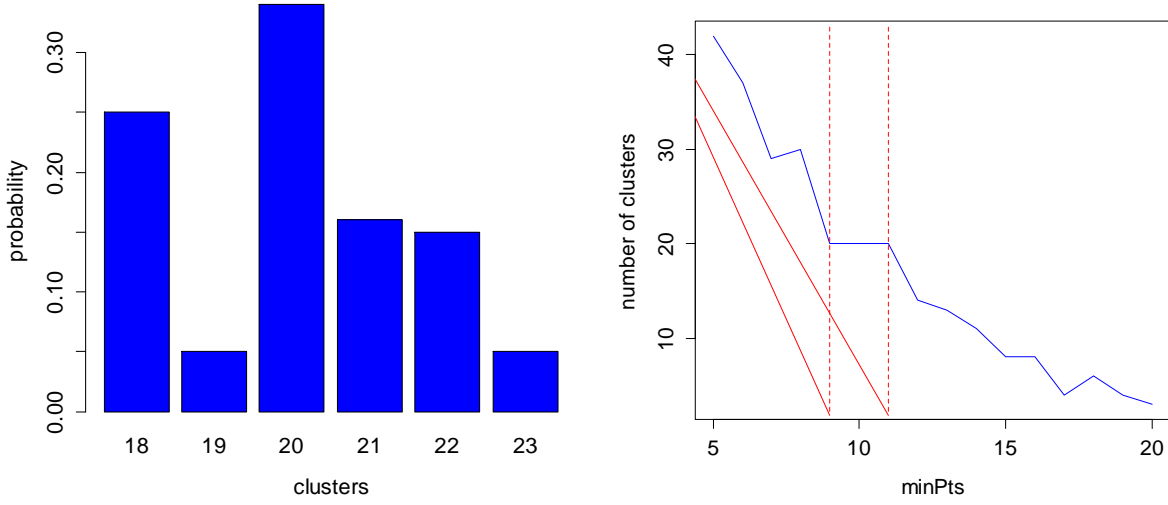


Figure 20: Number of topics estimation for the 20newsgroup dataset ($minPts = 10$)

The Ball index is correct in the WikiRef150 dataset, HDBSCAN and Dunn are correct in the WikiRef220 dataset and the indices DB, Silhouette, Dunn, SDindex and DP-means are correct in the WikiRef186 datasets. However, in all datasets, the estimation given by the majority vote over 10 realizations of the DBSCAN-Martingale coincides with the ground truth number of clusters. In Figures 17 to 20, we present the estimation of the number of clusters for 100 realizations of the DBSCAN-Martingale, in order to show that after 10 runs of 10 realizations the output of Algorithm 1 remains the same. The parameter $minPts$ is taken equal to 10 for the 20news dataset and 20 for all other cases.

Evaluation of the MULTISENSOR topic detection framework

The evaluation measure is the popular Normalized Mutual Information (NMI), mainly used for the evaluation of clustering techniques, which allows us to compare results when the number of outputted clusters does not match the number of clusters in the ground truth (Kullis and Jordan, 2012). For the output k of each method of Table 8, we show the average of 10 runs of LDA (and the corresponding standard deviation) in Table 9. For the WikiRef150 dataset, the combination of Ball index with LDA provides the highest NMI. For the WikiRef220 dataset, the combinations of HDBSCAN with LDA and Dunn index with LDA also provide the highest NMI. For the WikiRef186 dataset, the combinations of LDA with the indices DB, Silhouette, Dunn, SDindex and DP-means perform well. However, in all 4 datasets, the MULTISENSOR framework provides the highest NMI score and in the case of 20news dataset, the combination of DBSCAN-Martingale with LDA is the only method which provides the highest NMI score. Without using LDA, the best partition provided by the baseline methods of Table 8 has NMI less than 50% in all WikiRef150, WikiRef186 and WikiRef220 and therefore, is not presented. In contrast, we adopted the LDA method which achieves NMI scores up to 85.6%. Density-based algorithms such as HDBSCAN and DBSCAN-Martingale assigned too much noise in our datasets, a fact that affects the clustering performance, especially when compared to LDA in news clustering, thus we kept only \hat{k} .

Index + LDA	WikiRef150	WikiRef186	WikiRef220	20 news
CH	0.5537 \pm 0.0111	0.6080 \pm 0.0169	0.6513 \pm 0.0126	0.3073 \pm 0.0113
Duda	0.6842 \pm 0.0400	0.6469 \pm 0.0271	0.6381 \pm 0.0429	0.1554 \pm 0.0067
Pseudo t^2	0.6842 \pm 0.0400	0.6469 \pm 0.0271	0.6381 \pm 0.0429	0.1554 \pm 0.0067
C-index	0.5614 \pm 0.0144	0.6469 \pm 0.0271	0.6381 \pm 0.0429	0.1554 \pm 0.0067
Ptbiserial	0.6469 \pm 0.0283	0.6469 \pm 0.0271	0.8262 \pm 0.0324	0.3073 \pm 0.0113
DB	0.6842 \pm 0.0400	0.7892 \pm 0.0553	0.8262 \pm 0.0324	0.1554 \pm 0.0067
Frey	0.6842 \pm 0.0400	0.6469 \pm 0.0271	0.6381 \pm 0.0429	0.2460 \pm 0.0198
Hartigan	0.5887 \pm 0.0157	0.6513 \pm 0.0184	0.7156 \pm 0.0237	0.3126 \pm 0.0098
Ratkovsky	0.5866 \pm 0.0123	0.6201 \pm 0.0188	0.6570 \pm 0.0107	0.3073 \pm 0.0113
Ball	0.7687 \pm 0.0231	0.7655 \pm 0.0227	0.7601 \pm 0.0282	0.2101 \pm 0.0192
McClain	0.6842 \pm 0.0400	0.6469 \pm 0.0271	0.6381 \pm 0.0429	0.1554 \pm 0.0067
KL	0.6097 \pm 0.0232	0.6670 \pm 0.0156	0.7091 \pm 0.0257	0.3077 \pm 0.0094
Silhouette	0.5537 \pm 0.0111	0.7892 \pm 0.0553	0.8032 \pm 0.0535	0.1554 \pm 0.0067
Dunn	0.5805 \pm 0.024	0.7892 \pm 0.0553	0.8560 \pm 0.0397	0.2101 \pm 0.0192
SDindex	0.7007 \pm 0.0231	0.7892 \pm 0.0553	0.8262 \pm 0.0324	0.2101 \pm 0.0192
SDbw	0.5537 \pm 0.0111	0.7668 \pm 0.0351	0.8262 \pm 0.0324	0.2101 \pm 0.0192
NbClust	0.6842 \pm 0.0400	0.6469 \pm 0.0271	0.8262 \pm 0.0324	0.1554 \pm 0.0067
DP-means	0.7007 \pm 0.0231	0.7892 \pm 0.0553	0.8278 \pm 0.0341	0.3077 \pm 0.0094
HDBSCAN-EOM	0.7145 \pm 0.029	0.7630 \pm 0.0530	0.8560 \pm 0.0397	0.3106 \pm 0.0134
DBSCAN-Martingale (MULTISENSOR)	0.7687 \pm 0.0231	0.7892 \pm 0.0553	0.8560 \pm 0.0397	0.3137\pm 0.0130

Table 9: Normalized Mutual Information over 10 runs of LDA for each estimation of the number of topics, for the evaluation of the MULTISENSOR topic detection framework

4.5.3 Evaluation results in the MULTISENSOR database

We also applied our topic detection framework in the context of multilingual topic detection, on the retrieved results for a given query. In order to evaluate the clustering of the retrieved news articles, we use the average precision (AP), broadly used in the context of information retrieval, clustering and classification. A document d of a cluster C is considered relevant to C (true positive) if at least one concept associated with document d appears also in the label of cluster C . Precision is considered the fraction of relevant documents in a cluster and average precision is the average for all clusters of a query. Finally, we average the AP scores for all considered queries to obtain the Mean Average Precision (MAP).

The AP scores per query and the MAP scores per method are shown in Table 10. The estimation of the number of news clusters is presented in Table A6 in Appendix A. We observe an average increase of 9.65% in MAP, when the MULTISENSOR topic detection framework is compared to the second highest MAP score (by Hartigan+LDA) and an increase of 10.20%, when compared to the most recent approach (NbClust+LDA). The aforementioned results indicate that the topic detection task has been fully achieved,

according to the indicators defined in deliverable D1.2 (Self-assessment plan v2), as we have an average increase in MAP of more than 5%, compared to the baseline approach.

Index + LDA	energy crisis	energy policy	home appliances	solar energy	MAP
CH	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
Duda	0.4498 \pm 0.0671	0.5534\pm0.0457	0.4299 \pm 0.0237	0.4484 \pm 0.0067	0.4703
Pseudo t^2	0.4498 \pm 0.0671	0.5534\pm0.0457	0.4299 \pm 0.0237	0.4484 \pm 0.0067	0.4703
C-index	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
Ptbiserial	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
DB	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
Frey	0.3541 \pm 0.0181	0.3911 \pm 0.0033	0.3745 \pm 0.064	0.4484 \pm 0.0067	0.3920
Hartigan	0.5938 \pm 0.0502	0.5336 \pm 0.0375	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5794
Ratkovsky	0.5357 \pm 0.0151	0.5371 \pm 0.0357	0.4962 \pm 0.0721	0.5375 \pm 0.0446	0.5266
Ball	0.4207 \pm 0.0093	0.4501 \pm 0.0021	0.4975 \pm 0.016	0.4464 \pm 0.0614	0.4536
McClain	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.3745 \pm 0.064	0.5961 \pm 0.0347	0.5215
KL	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5701 \pm 0.0145	0.5961 \pm 0.0347	0.5704
Silhouette	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
Dunn	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
SDindex	0.3541 \pm 0.0181	0.3911 \pm 0.0033	0.5942 \pm 0.0282	0.4484 \pm 0.0067	0.4469
SDbw	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
NbClust	0.5786 \pm 0.0425	0.5371 \pm 0.0357	0.5942 \pm 0.0282	0.5961 \pm 0.0347	0.5765
DP-means	0.3541 \pm 0.0181	0.3911 \pm 0.0033	0.3745 \pm 0.064	0.4484 \pm 0.0067	0.3920
HDBSCAN-EOM	0.4498 \pm 0.0671	0.3911 \pm 0.0033	0.5951 \pm 0.0184	0.5375 \pm 0.0446	0.4933
DBSCAN-Martingale (MULTISENSOR)	0.7691\pm0.0328	0.5534\pm0.0457	0.6115\pm0.0225	0.6073\pm0.0303	0.6353

Table 10: Average Precision (\pm standard deviation) over 10 runs of LDA for each estimation of the number of topics, for the evaluation of the MULTISENSOR topic detection framework

Regarding the time performance of the DBSCAN-Martingale, we selected several baseline approaches to compare their processing time with our approach. In Figure 21, the number of news clusters is estimated for $T = 5$ iterations for the DBSCAN-Martingale and for maximum number of clusters set to 15 for the indices Duda, Pseudo t^2 , Silhouette, Dunn and SDindex in the “NbClust” package in R (Charrad et al., 2014). We observe that the DBSCAN-Martingale, which uses the “dbscan” package (Hahsler, 2015), is faster than all other methods and even when it is applied on 500 documents, it is able to reach a decision about the number of clusters approximately in 0.4 seconds.

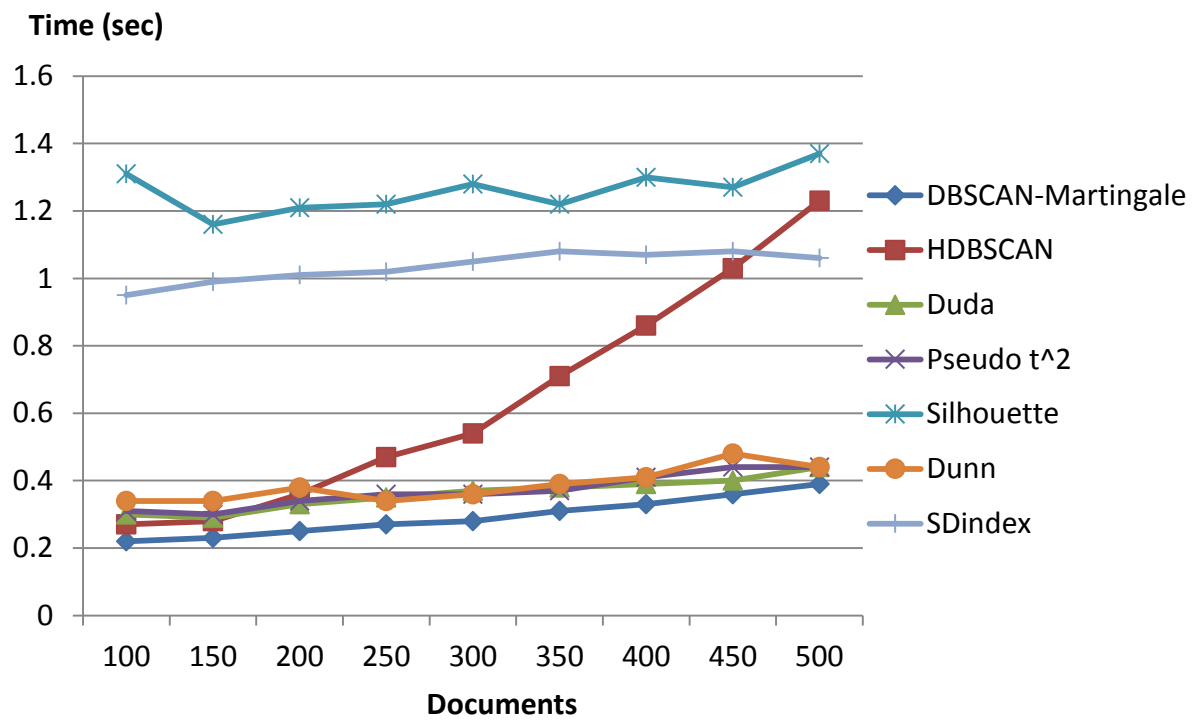


Figure 21: Time Performance of several methods to estimate the number of news clusters

5 CONCLUSIONS

A novel framework for multimedia retrieval which is based on textual concepts, visual features and visual concepts has been presented. However, a more general approach which fuses M modalities is also discussed. The overall multimedia retrieval framework is, moreover, language independent, due to the language-agnostic modalities involved, so the MULTISENSOR multimedia retrieval framework is multimodal and multilingual. The multimedia indexing SIMMO model has been updated and offers faster response to cross-modal queries, i.e. from text to image and vice versa, within webpage IDs. This model is populated using the features from WP2 and WP3 and therefore it is domain and language independent.

We have presented a hybrid framework for topic detection, based on the DBSCAN-Martingale for the estimation of the number of news clusters, followed by the assignment of news articles to topics using Latent Dirichlet Allocation. The novel method for estimating the number of topics, based on language independent features, combined with LDA has been tested as a news clustering framework and performs very well, as evaluated using the Normalized Mutual Information. The MULTISENSOR topic detection framework is multilingual and multimodal, since it fuses more than one sources of information of the same multimedia object, such as concepts and named entities. The fact that the MULTISENSOR topic detection module relies on language independent features makes it easily adaptable to other languages. Moreover, the module is based on an unsupervised technique and therefore it is domain independent.

Finally, we have presented the final framework for the category-based classification task, in which, compared to the initial framework that was described in D4.1, the visual modality is replaced by a textual-based one. This textual-based modality relies on a recently proposed methodology for word embeddings called word2vec. The use of only textual-based modalities (N-gram and word2vec), coupled with late fusion strategies that are based on the operational capabilities of RF, provides a very efficient classification approach, as demonstrated by the experiments that were conducted within MULTISENSOR. Finally, it should be noted that while it is possible to extract N-gram textual features and word2vec feature vectors from texts in all languages, NLP (Natural Language Processing) preprocessing is actually a language specific task. Provided that the right tools and resources are available (e.g. text corpora and stopword lists in languages other than English etc.), it is feasible to adapt the MULTISENSOR category-based classification module to other languages/domains (details on resources required for such an adaptation will be together with the final exploitation plans in D9.7).

6 REFERENCES

- Aggarwal, C. C., & Zhai, C. 2012. "A survey of text clustering algorithms", In *Mining Text Data*, pp. 77-128, Springer US.
- Allan, J. (Ed.) 2012. "Topic detection and tracking: event-based information organization", vol. 12, Springer Science & Business Media.
- Ankerst, M., Breunig, M. M., Kriegel, H. P., & Sander, J. 1999. "OPTICS: ordering points to identify the clustering structure", In *ACM Sigmod Record*, vol. 28(2), pp. 49-60, ACM.
- Ah-Pine, J., Csurka, G., & Clinchant, S. 2015. "Unsupervised Visual and Textual Information Fusion in CBMIR Using Graph-Based Methods", *ACM Transactions on Information Systems (TOIS)*, vol. 33(2), 9.
- Atrey, P. K., Hossain, M. A., El Saddik, A., & Kankanhalli, M. S. 2010. "Multimodal fusion for multimedia analysis: a survey", *Multimedia systems*, vol. 16(6), pp. 345-379.
- Ball, G. H., & Hall, D. J. 1965. "*ISODATA, a novel method of data analysis and pattern classification*", Stanford Research Institute, Menlo Park. (NTIS No. AD 699616).
- Blei, D. M., & Jordan, M. I. 2003. "Modeling annotated data", In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 127-134, ACM.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. 2003. "Latent dirichlet allocation", *the Journal of machine Learning research*, vol. 3, pp. 993-1022.
- Breiman, L. 2001. "Random Forests", In *Machine Learning*, vol. 45(1), pp. 5-32.
- Campello, R. J., Moulavi, D., & Sander, J. 2013. "Density-based clustering based on hierarchical density estimates", In *Advances in Knowledge Discovery and Data Mining*, pp. 160-172, Springer Berlin Heidelberg.
- Caliński, T., & Harabasz, J. 1974. "A dendrite method for cluster analysis", *Communications in Statistics-theory and Methods*, vol. 3(1), pp. 1-27.
- Charrad, M., Ghazzali, N., Boiteau, V., & Niknafs, A. 2014. "NbClust: an R package for determining the relevant number of clusters in a data set", *Journal of Statistical Software*, vol. 61(6), pp. 1-36.
- Costa Pereira, J., Coviello, E., Doyle, G., Rasiwasia, N., Lanckriet, G. R., Levy, R., & Vasconcelos, N. 2014. "On the role of correlation and abstraction in cross-modal multimedia retrieval", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36(3), pp. 521-535.
- Couture-Beil, A. 2014. "rjson: JSON for R", R package version 0.2.15. <http://CRAN.R-project.org/package=rjson>
- Daiber, J., Jakob, M., Hokamp, C., & Mendes, P. N. 2013. "Improving efficiency and accuracy in multilingual entity extraction", In *Proceedings of the 9th International Conference on Semantic Systems*, pp. 121-124, ACM.
- Davies, D. L., & Bouldin, D. W. 1979. "A cluster separation measure", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2), pp. 224-227.

- Duda, R. O., & Hart, P. E. 1973. *“Pattern classification and scene analysis”*, vol. 3, New York: Wiley.
- Dunn, J. C. 1974. “Well-separated clusters and optimal fuzzy partitions”, *Journal of cybernetics*, vol. 4(1), pp. 95-104.
- Doob, J. L. 1953. *“Stochastic processes”*, vol. 101, Wiley: New York.
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. 1996. “A density-based algorithm for discovering clusters in large spatial databases with noise”, In *Kdd*, vol. 96, no. 34, pp. 226-231.
- Feinerer, I. & Hornik, K. 2015. “tm: Text Mining Package”, R package version 0.6-2, <http://CRAN.R-project.org/package=tm>
- Frey, T., & Van Groenewoud, H. 1972. “A cluster analysis of the D2 matrix of white spruce stands in Saskatchewan based on the maximum-minimum principle”, *The Journal of Ecology*, pp. 873-886.
- Grubinger, M., Clough, P., Müller, H., & Deselaers, T. 2006. “The IAPR-TC12 benchmark: A new evaluation resource for visual information systems”, In *International Workshop OntoImage*, pp. 13-23.
- Gruen, B. & Hornik, K. 2011. “topicmodels: An R Package for Fitting Topic Models”, *Journal of Statistical Software*, vol. 40(13), pp. 1-30, <http://www.jstatsoft.org/v40/i13/>
- Hafner, J., Sawhney, H. S., Equitz, W., Flickner, M., & Niblack, W. 1995. “Efficient color histogram indexing for quadratic form distance functions”, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17(7), pp. 729-736.
- Halkidi, M., Vazirgiannis, M., & Batistakis, Y. 2000. “Quality scheme assessment in the clustering process”, In *Principles of Data Mining and Knowledge Discovery*, pp. 265-276, Springer Berlin Heidelberg.
- Hartigan, J. A. 1975. *“Clustering algorithms”*, New York: John Wiley & Sons.
- Hahsler, M. 2015. “dbscan: Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms”, R package version 0.9-5, <http://CRAN.R-project.org/package=dbscan>
- Hsu, W. H., Kennedy, L. S., & Chang, S. F. 2007. “Video search reranking through random walk over document-level context graph”, In *Proceedings of the 15th international conference on Multimedia*, pp. 971-980, ACM.
- Hubert, L., & Arabie, P. 1985. “Comparing partitions”, *Journal of classification*, vol. 2(1), pp. 193-218.
- Hubert, L. J., & Levin, J. R. 1976. “A general statistical framework for assessing categorical clustering in free recall”, *Psychological bulletin*, vol. 83(6), 1072.
- Jégou, H., Douze, M., Schmid, C., & Pérez, P. 2010. “Aggregating local descriptors into a compact image representation”, In *IEEE Conference on Computer Vision & Pattern Recognition*, pp. 3304-3311.
- Jeon, J., Lavrenko, V., & Manmatha, R. 2003. “Automatic image annotation and retrieval using cross-media relevance models”, In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 119-126, ACM.

- Ju, R., Zhou, P., Li, C. H., & Liu, L. 2015. "An Efficient Method for Document Categorization Based on Word2vec and Latent Semantic Analysis", In *Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM)*, 2015 IEEE International Conference on, pp. 2276-2283, IEEE.
- Kaufman, L., & Rousseeuw, P. J. 1990. "Finding groups in data. an introduction to cluster analysis", *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics*, New York: Wiley, 1990, 1.
- Krzanowski, W. J., & Lai, Y. T. 1988. "A criterion for determining the number of groups in a data set using sum-of-squares clustering", *Biometrics*, pp. 23-34.
- Kulis, B., & Jordan, M. I. 2012. "Revisiting k-means: New algorithms via Bayesian nonparametrics", *arXiv preprint arXiv:1111.0352*.
- Kumar, A., & Daumé, H. 2011. "A co-training approach for multi-view spectral clustering", In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 393-400.
- Lilleberg, J., Zhu, Y., & Zhang, Y. 2015. "Support vector machines and Word2vec for text classification with semantic features", In *Cognitive Informatics & Cognitive Computing (ICCI* CC)*, 2015 IEEE 14th International Conference on, pp. 136-140, IEEE.
- Mai, S. T., He, X., Feng, J., Plant, C., & Böhm, C. 2014. "Anytime density-based clustering of complex data", *Knowledge and Information Systems*, pp. 1-37.
- McClain, J. O., & Rao, V. R. 1975. "Clustisz: A program to test for the quality of clustering of a set of objects", *JMR, Journal of Marketing Research (pre-1986)*, 12(000004), 456.
- Mei, T., Rui, Y., Li, S., & Tian, Q. 2014. "Multimedia search reranking: A literature survey", *ACM Computing Surveys (CSUR)*, vol. 46(3), 38.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. 2013. "Efficient estimation of word representations in vector space", *arXiv preprint arXiv:1301.3781*.
- Milligan, G. W., & Cooper, M. C. 1985. "An examination of procedures for determining the number of clusters in a data set", *Psychometrika*, vol. 50(2), pp. 159-179.
- Porter, M. F. 1980. "An algorithm for suffix stripping", *Program*, vol. 14(3), pp. 130-137.
- Qian, M., & Zhai, C. 2014. "Unsupervised feature selection for multi-view clustering on text-image web news data", In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pp. 1963-1966, ACM.
- Rasiwasia, N., Costa Pereira, J., Coviello, E., Doyle, G., Lanckriet, G. R., Levy, R., & Vasconcelos, N. 2010. "A new approach to cross-modal multimedia retrieval", In *Proceedings of the international conference on Multimedia*, pp. 251-260, ACM.
- Ratkowsky, D. A., & Lance, G. N. 1978. "A criterion for determining the number of groups in a classification", *Australian Computer Journal*, vol. 10(3), pp. 115-117.
- Safadi, B., & Quénot, G. 2011. "Re-ranking by local re-scoring for video indexing and retrieval", In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 2081-2084, ACM.

- Safadi, B., Sahuguet, M., & Huet, B. 2014. "When textual and visual information join forces for multimedia retrieval", In *Proceedings of International Conference on Multimedia Retrieval*, pp. 265, ACM.
- Sander, J., Qin, X., Lu, Z., Niu, N., & Kovarsky, A. 2003. "Automatic extraction of clusters from hierarchical clustering representations", In *Advances in knowledge discovery and data mining*, pp. 75-87, Springer Berlin Heidelberg.
- Schneider, J., & Vlachos, M. 2013. "Fast parameterless density-based clustering via random projections", In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pp. 861-866, ACM.
- Siddiquie, B., White, B., Sharma, A., & Davis, L. S. 2014. "Multi-Modal Image Retrieval for Complex Queries using Small Codes", In *Proceedings of International Conference on Multimedia Retrieval*, pp. 321, ACM.
- Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. 2006. "Hierarchical dirichlet processes", *Journal of the american statistical association*, vol. 101(476).
- Tsikrika, T., Andreadou, K., Moumtzidou, A., Schinas, E., Papadopoulos, S., Vrochidis, S., & Kompatsiaris, I. 2015. "A Unified Model for Socially Interconnected Multimedia-Enriched Objects", In *MultiMedia Modeling*, pp. 372-384, Springer International Publishing.
- Tsikrika, T., & Kludas, J. 2010. "The Wikipedia image retrieval task", In *ImageCLEF*, pp. 163-183, Springer Berlin Heidelberg.
- Van De Sande, K. E., Gevers, T., & Snoek, C. G. 2010. "Evaluating color descriptors for object and scene recognition", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32(9), pp. 1582-1596.
- Wang, J., He, Y., Kang, C., Xiang, S., & Pan, C. 2015. "Image-Text Cross-Modal Retrieval via Modality-Specific Feature Learning", In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 347-354, ACM.
- Wang, Y., Lin, X., & Zhang, Q. 2013. "Towards metric fusion on multi-view data: a cross-view based graph random walk approach", In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pp. 805-810, ACM.
- Wang, Y., Wu, F., Song, J., Li, X., & Zhuang, Y. 2014. "Multi-modal mutual topic reinforce modeling for cross-media retrieval", In *Proceedings of the ACM International Conference on Multimedia*, pp. 307-316, ACM.
- Xing, C., Wang, D., Zhang, X., & Liu, C. 2014. "Document classification with distributions of word vectors", In *Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA)*, pp. 1-5, IEEE.
- Xu, S., Li, H., Chang, X., Yu, S. I., Du, X., Li, X., Jiang, L., Mao, Z., Lan, Z., Burger, S. & Hauptmann, A. 2015. "Incremental Multimodal Query Construction for Video Search", In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 675-678, ACM.
- Zhang, D., Xu, H., Su, Z., & Xu, Y. 2015. "Chinese comments sentiment classification based on word2vec and SVM perf", *Expert Systems with Applications*, vol. 42(4), pp. 1857-1863.

Zhou, Q., Hong, W., Luo, L., & Yang, F. 2010. "Gene selection using random forest and proximity differences criterion on DNA microarray data", Journal of Convergence Information Technology, vol. 5(6), pp. 161-170.

A Appendix

A.1 Confusion matrices (topic-based classification)

Predicted Observed	Economy- Business- Finance	Health	Lifestyle- Leisure	Nature- Environment	Politics	Science- Technology
Economy- Business- Finance	1091	0	26	8	0	17
Health	20	47	18	0	0	15
Lifestyle- Leisure	90	4	962	7	2	58
Nature- Environment	35	0	6	291	1	19
Politics	68	0	12	8	81	6
Science- Technology	102	1	72	9	2	545

Table A1: Test set confusion matrix (N-gram RF model)

Predicted Observed	Economy- Business- Finance	Health	Lifestyle- Leisure	Nature- Environment	Politics	Science- Technology
Economy- Business- Finance	1018	0	61	14	12	37
Health	8	42	27	1	2	20
Lifestyle- Leisure	83	3	952	6	7	72
Nature-	27	0	22	283	3	17

Environment						
Politics	36	0	12	13	109	5
Science-Technology	93	10	126	11	4	487

Table A2: Test set confusion matrix (word2vec RF model)

Predicted Observed	Economy-Business-Finance	Health	Lifestyle-Leisure	Nature-Environment	Politics	Science-Technology
Economy-Business-Finance	1073	0	36	8	3	22
Health	15	46	22	1	1	15
Lifestyle-Leisure	72	2	986	7	1	55
Nature-Environment	31	0	6	295	2	18
Politics	48	0	10	13	102	2
Science-Technology	100	5	89	9	1	527

Table A3: Test set confusion matrix (RF late fusion – Weighting based on OOB error per topic)

Predicted Observed	Economy-Business-Finance	Health	Lifestyle-Leisure	Nature-Environment	Politics	Science-Technology
Economy-Business-Finance	1062	0	33	9	7	31

Health	10	48	23	1	1	17
Lifestyle- Leisure	70	1	985	7	2	58
Nature- Environment	33	0	6	293	2	18
Politics	47	0	9	12	105	2
Science- Technology	96	5	89	10	2	529

Table A4: Test set confusion matrix (RF late fusion – Weighting based on proximity ratio per topic)

Predicted Observed	Economy- Business- Finance	Health	Lifestyle- Leisure	Nature- Environment	Politics	Science- Technology
Economy- Business- Finance	1064	0	32	9	7	30
Health	11	47	24	1	1	16
Lifestyle- Leisure	70	2	984	7	2	58
Nature- Environment	33	0	6	293	2	18
Politics	47	0	9	12	105	2
Science- Technology	97	5	85	10	3	531

Table A5: Test set confusion matrix (RF late fusion – Weighting based on adjusted proximity ratio per topic)

A.2 Estimation of the number of news clusters (topic-event detection)

Index	energy crisis	energy policy	home appliances	solar energy
CH	12	8	15	15
Duda	4	4	3	2
Pseudo t^2	4	4	3	2
C-index	12	8	15	15
Ptbiserial	12	8	15	15
DB	12	8	15	15
Frey	2	2	2	2
Hartigan	11	7	15	15
Ratkowsky	7	8	5	5
Ball	3	3	3	3
McClain	12	8	2	15
KL	12	8	11	15
Silhouette	12	8	15	15
Dunn	12	8	15	15
SDindex	2	2	15	2
SDbw	12	8	15	15
NbClust	12	8	15	15
DP-means	2	2	2	2
HDBSCAN-EOM	4	2	10	5
DBSCAN-Martingale	6	4	9	10

Table A6: Estimation of the number of topics in the MULTISENSOR queries